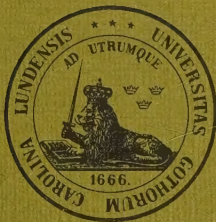


CONGESTION CONTROL IN PACKET SWITCHING COMPUTER COMMUNICATION NETWORKS

ULF KÖRNER



**DEPARTMENT OF TELECOMMUNICATION SYSTEMS
LUND INSTITUTE OF TECHNOLOGY
LUND SWEDEN**

1982.

CONGESTION CONTROL IN PACKET SWITCHING

COMPUTER COMMUNICATION NETWORKS

by

Ulf Körner

Department of Telecommunication Systems

Lund Institute of Technology

LUTDEX/(TETS-1001)/1-295/(1982)



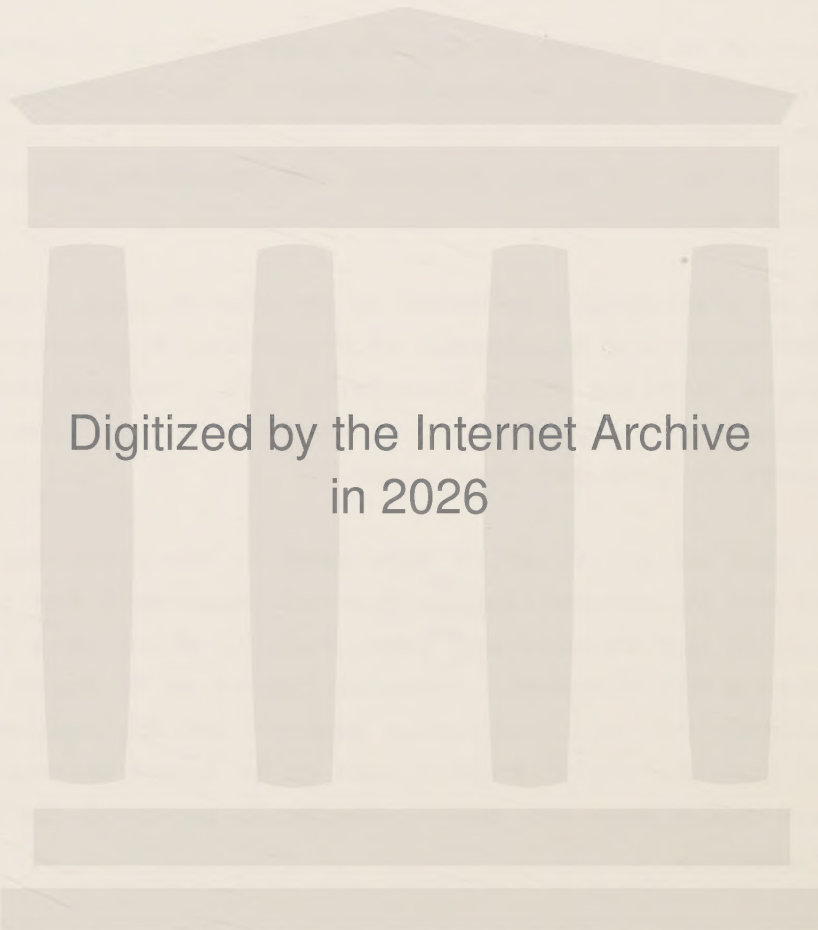
ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my advisor Professor Bengt Wallström, Head of the Department of Telecommunication Systems, Lund Institute of Technology, for his help, guidance and friendship during this work.

I am also deeply indebted to Dr Arne Nilsson, Professor at the Department of Electrical Engineering North Carolina State University, USA, for his ever encouraging support and guidance during the time he spent in Lund and ever since.

A dept of gratitude is also owed to the personnel of the Telecommunication Systems Department for providing the stimulating atmosphere in which this research was performed. Special thanks go to Berth Eklundh for his invaluable remarks and for improving my English style. Thanks also go to Bengt Stavenow and David Rapp for their remarks on parts of this research.

I am deeply indebted to Monica Bäckström for typing my manuscript and for her ever encouraging enthusiasm. Special thanks go to Elzbieta Szybicka-Strauss and to Lena Svensson for drawing all the figures.



Digitized by the Internet Archive
in 2026

ABSTRACT

The objective of this research is in general to develop analytic tools for the performance study of computer communication networks, and is in particular motivated by a need to understand how restricted buffer sharing policies can be used for congestion control in the nodes of a packet switching computer communication network.

Models of a single packet switching node with finite storage and restricted buffer sharing policies are developed. Effects of acknowledgement signals as well as retransmission are considered.

We introduce the concept "penalty functions" and incorporate these in the packet switching node model in order to give priority to packets that have reached their destination node and thus should not be lost.

The single packet switching node models are used to form a model of a complete packet switching network.

We present for the different models comprehensive results as throughput, delays, blocking probabilities and buffer allocation schemes.

We derive a number of analytic tools, introduce some new concepts and clarify some techniques, used in our models, to compute the main performance characteristics.

CONTENTS

	page
Chapter 1 INTRODUCTION	1
1.1 Summery of purpose	1
1.2 A brief survey of the research	2
Chapter 2 COMPUTER COMMUNICATION NETWORKS	4
2.1 Introduction	4
2.2 Towards distributed computer networks	4
2.3 Network structures	6
2.4 Protocols	10
2.5 Different sub network topologies	13
2.6 A brief survey of existing packet switching networks	15
2.7 Interconnection of networks	19
2.8 Computer networks in the eighties	22
Chapter 3 QUEUEING AND QUEUEING NETWORK THEORIES	25
3.1 Queueing theory - a unique tool for under- standing the behaviour of complex communication systems	25
3.2 Definitions	26
3.3 The M/M/1 system	29
3.4 The G/G/1 system	34
3.5 Queueing Network Theory	35
3.6 The exponential assumption	37

Chapter 4	FLOW CONTROL - CONGESTION CONTROL	41
4.1	Introduction	41
4.2	Negativ effects in uncontrolled packet switching networks	41
4.3	Definitions	45
4.4	Flow control mechanisms	46
4.5	Conclusions	48
Chapter 5	RESTRICTED BUFFER SHARING POLICIES	49
5.1	Introduction	49
5.2	Some basic restricted buffer sharing policies	52
5.3	Problems when implementing restricted buffer sharing policies in a net	58
5.4	Comparisons	59
5.5	Previous analyses of restricted buffer sharing policies	60
Chapter 6	A PACKET SWITCHING NODE MODEL	61
6.1	Introduction	61
6.2	The packet switching node	61
6.3	The problem in essence	64
6.4	The steady state probability distribution	66
6.5	Blocking probabilities	67
6.6	Cost functions	70
6.6.1	Cost functions - of what?	70
6.6.2	The square-root rule	71
6.7	Sample results	75
6.7.1	Blocking probabilities	76
6.7.2	Buffer allocation	89
6.7.3	Cost reduction	96

6.7.4	Throughput	99
6.7.5	Delays	104
6.7.6	FSMA versus CS	109
6.7.7	Utilization and Maximum Queue lengths	111
6.8	The impact of offered load and arrival rate	115
6.9	Conclusions	123
Chapter 7	A RECURSIVE TECHNIQUE TO COMPUTE THE NORMALIZATION CONSTANT, THE BLOCKING PROBABILITIES AND THE MEAN NUMBER OF PACKETS	126
7.1	Introduction	126
7.2	Feasible states within the n-dimensional state space	126
7.3	The normalization constant	129
7.4	Blocking probabilities	133
7.5	The mean number of packets	134
7.6	Conclusions	135
Chapter 8	A PACKET SWITCHING NODE MODEL WITH ACKNOWLEDGEMENT SIGNALS AND RETRANSMISSION	136
8.1	Introduction	136
8.2	A model with acknowledgement signals	136
8.2.1	The steady state distribution	138
8.2.2	Performance Measures	139
8.2.3	Sample results	141
8.3	A model with retransmission	152
8.3.1	Unlimited number of buffers	152
8.3.2	A limited number of buffers	154
8.3.3	The general case - n transmission links	155
8.3.4	Sample results	155
8.4	Conclusions	160

Chapter 9	USING PENALTY FUNCTIONS WHEN OPTIMIZING THE NUMBER OF DEDICATED BUFFERS	162
9.1	Introduction	162
9.2	Penalty functions	162
9.3	Sample results	164
9.4	Conclusions	174
Chapter 10	A MODEL OF A PACKET SWITCHING NETWORK	175
10.1	Introduction	175
10.2	Basic assumptions	176
10.3	Sample Networks	178
10.4	The Problem in essence	182
10.4.1	Algorithms to compute link flows when blocking is known	182
10.4.2	An iterative algorithm to compute linkflow	185
10.4.3	Convergence and length of iteration	186
10.5	Algorithms to compute end to end blocking probabilities and delays	188
10.5.1	Matrix algorithms	191
10.5.2	Little's formula applied to a network with loss	196
10.6	Sample results	200
10.6.1	The 6-node net	201
10.6.1.1	Offered load	201
10.6.1.2	The number of lost packets	204
10.6.1.3	Throughput	205
10.6.2	The 11-node net	208
10.6.2.1	Offered load	208
10.6.2.2	The number of lost packets	211
10.6.2.3	Throughput	213
10.6.2.4	Blocking probabilities	221
10.6.2.5	Delays	224
10.6.2.6	The mean number of packets in the net	226
10.7	Conclusions	228

Chapter 11	CONCLUSIONS AND FURTHER RESEARCH	230
11.1	Conclusions	230
11.2	Further research	231
APPENDIX A		233
A.1	Introduction	233
A.2	The steady state probability distribution of a limited two dimensional birth-death process	233
A.2.1	The global balance equations	236
A.2.2	The z-transform	237
A.2.3	Inversion of the z-transform	250
A.3	The Steady state distribution corresponding to a node with acknowledgement signals	254
A.3.1	The global balance equations	255
A.3.2	The z-transform	257
A.4	Acknowledgement signals and retransmission	264
A.4.1	The global balance equations	264
A.4.2	The z-transform	265
APPENDIX B		274
B.1	Introduction	274
B.2	Utilization under the CS policy	274
B.3	Utilization under the SMA policy	276
REFERENCES		285

CHAPTER 1 - INTRODUCTION

1.1 Summery of purpose

During the last decade we have witnessed a growth in packet switching computer communication networks. This evolution has been guided by lessons learned in the early networks designs during the sixties. Today we find a number of small local as well as large public packet switching networks spread over the world. One of the main reasons for the rise of packet switching networks is the considerable economical benefits that can be achieved by sharing resources, i e by sharing computational capacities as well as transmission links and switching computers.

This rather new technique has brought about the need for analytic tools to understand the network behaviour when in position of a network design. Great efforts have been made during the last years to develop such tools. However, the desire to understand how these complex systems behave, how they should be controlled from congestion etc requires further extensive efforts in developing new analytic tools.

In general, the goal of this research is to develop analytic tools for the performance study of computer communication networks, and is in particular motivated by a need to understand how restricted buffer sharing policies can be used for congestion control in the nodes of a packet switching computer communication network. Though our models sometimes are rather simplistic, they can make us understand a basic network behaviour

1.2 A brief survey of the research

In chapter 2 we give a brief survey of the computer communication area. The two main switching techniques: circuit switching and packet switching are presented. Their benefits and drawbacks are discussed.

Chapter 3 contains our basic analytic tools: queueing and queueing network theory. We present wellknown performance characteristics as mean number of packets in a server system, average response time and blocking probabilities. We conclude the chapter with a discussion on the credibility of exponential assumptions.

In chapter 4 we discuss the effects of flow control and congestion control mechanisms in packet switching networks. We present a number of known techniques of how to implement these mechanisms.

Chapter 5 presents the restricted buffer sharing policies that we will use in our packet switching node models. The policies are used in a network in order to improve performance characteristics and to protect the entire network from degradation in abnormal and heavy load situations.

Chapter 6 deals with a model of a packet switching node with finite storage and restricted buffer sharing policies. We compare how different policies effect the main performance characteristics.

In chapter 7 we derive a recursive technique to compute the main packet switching node performance characteristics.

In chapter 8 we extend our packet switching node model to incorporate the effects of acknowledgement signals

and retransmission. We find the same benefits of restricted buffer sharing policies as we found in chapter 6.

In chapter 9 we introduce the concept "penalty functions". Penalty functions are used when we optimize the restricted buffer sharing policies in order to give priority to packets that have reached their destination node and thus should not be lost.

Chapter 10 deals with a model of a packet switching network. A number of algorithms is derived to compute the main performance characteristics of a network. Results are presented for two sample networks, one 6-node net and one 11-node net.

In chapter 11 we finally make conclusions and make some suggestions for further research.

Appendix A contains a derivation of the steady state probability distributions for the models in chapter 6 and chapter 8.

In appendix B we derive some limiting values of the main performance characteristics.

CHAPTER 2 - COMPUTER COMMUNICATION NETWORKS

2.1 Introduction

During the last ten years the growth in computer communication networks has been intensive. The need for small local, nation wide and world wide, both private and public, computer communication networks is enormous. Eurodata '79, which is a study of data communication in Western Europe 1979-1987 [65], shows that almost 10% of all data communications traffic originating in Western Europe was international and the total amount of transactions, both national and international, was about 140 millions in the average working day in early 1979. A careful estimation gives a value of about 800 million transactions a day in 1987, out of which more than 100 millions would be international. A transaction is defined as a sequence of messages exchanged between two data communication users, which completes a task.

2.2 Towards distributed computer networks

Let us very briefly recapitulate some of the steps taken in the evolution of the computer communication field during the last 25 years. A long time ago, when programmers or scientists, located in different places, wished to use a central computer, telex- and telephone connections were established in an off-line mode. Thus data was sent for instance over the telex network and was punched on a paper tape punch at the computer central and finally, after being read by the paper type reader, the data had reached the computer. The transmission rates varied from that of the telex, say 5 characters per second, to about 75-1200 characters per second for a telephone connection.

The early remote job entry implementations, which as the just described principle were used in batch, do, concerning two points, differ from the former principle. The telephone link was connected via a modem directly to the computer and we had thus an on-line connection. The user had also some I/O devices in terms of a card-reader and a line-printer at his location. Data was sent from the card-reader via the telephone line to the computer, where it, perhaps after having waited for other batch jobs, finally was computed. The potential results were sent back and listed on the line-printer. The local scientists did really have a direct access to a computer. However, in many cases the batch system did contribute to very long access time.

The demands for a very fast access to the computer, in for instance transaction processing, called for a more flexible computer system, a system in which the access times were positively correlated to the execution times. During this period, the early and mid sixties, time-sharing systems were developed.

At this time, however, we still had just one computer surrounded by a lot of terminals, frequently so many that improvement of line utilization became interesting. When traffic flow between a user, or a group of users, and the computer reached a certain level, it could be profitable to lease a special line from the telephone administration. Thereby the availability of the connection usually increased. Furthermore concentrators and front-end processors were put into the system. Intelligent terminals made it to some extent possible to treat data before it was sent and to remove redundant information.

Among other things the need for resource sharing, i e sharing computer facilities, and not only sharing, but to use several computers, brought about the development of distributed computer networks. The first challenging efforts were made during the sixties, often at a major research laboratory. The need for sharing resources led to networks consisting of a distributed network of nodes and links and with terminals and computers attached to the nodes.

During the seventies many different kinds of networks were built. The lack of accepted standards made the group of networks very inhomogeneous. Thus, questions regarding connections of networks have, during the last five years, been vital. Today many networks of different kinds are connected and the work of standardization, regarding both the networks themselves and the interconnection, has advanced.

2.3 Network structures

Computer communication networks may conveniently be divided into two types: circuit switching and message switching. The distinction concerns primarily the sub network, i e the nodes or the switching computers and the high speed transmission links between the nodes, see figure 2.1. The sub network is usually easy to distinguish in a major public network, i e a network provided by for instance a telecommunication administration. In a small local network, however, the boundaries between the sub network and the host computers can be a little more troublesome to identify. For instance, in some networks the sub network contains just one transmission channel and no nodes and all the host computers are connected to this single transmission channel.

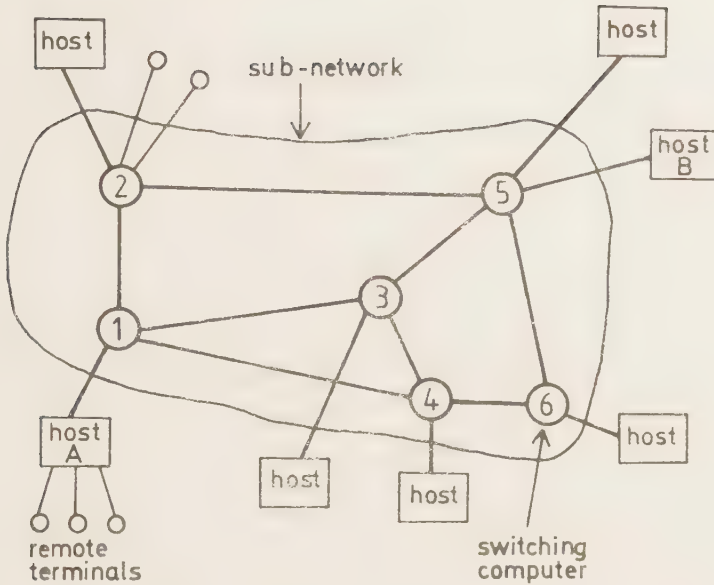


Figure 2.1

In a sense, the group of circuit switching networks are much more homogeneous than that of the message switching networks. Message switching networks are frequently based on both the store-and-forward principle and the multi-access principle.

In a circuit switching network (sometimes the notation line switching is used instead of circuit switching) a physical path is set up between the origin host and the destination host according to the same principles as in a telephone communication network. Thus the complete circuit is set up by a signalling message that, according to a specific routing procedure, finds its way from the origin host to the destination host. Once the path is set up, the network provides the physical

path to the user, independent of whether he uses it or not, and only when the user releases the path, the channel resources within the path are available to other users.

In a message switching network [4],[52], with a subnet that consists of nodes and transmission links and based on the store-and-forward principle, a header is attached to the information that is to be sent from one host computer to another. The header contains information regarding the destination host, the origin host, the length of the message, error control etc, and forms, together with the information that should be transmitted, a message. The entire message is first sent from the origin host to the node to which the host is connected. In this node the message is stored and a routing procedure chooses, according to the destination of the message and in some networks according to the load situation in the network, an appropriate outgoing transmission link. The message is stored in the node until the chosen transmission link becomes free and is then transmitted to the next node in the circuit path between the origin and the destination host. In this node the same procedure is repeated and the message is transmitted through the net according to the store-and-forward principle. When a message is stored in a node the just used transmission link could be used by other messages.

Packet switching networks [54] use the same principle as message switching networks. The information, however, that should be sent from one host computer to another, is divided into packets of fixed length. Each packet is given a header and finds its way through the network in a store-and-forward manner.

Packet switching networks, like the ALOHA network [1] and networks built on the Ethernet principle [7], use

a multiple access technique. Here the entire sub network consists of one single transmission channel, for instance a fast satellite link, which could be used by one user at a time,[44].

Many attempts have been made to compare the two principles circuit switching and packet switching. An overall comparison, in order to determine which is the best in general, is almost impossible to make. Let us make just a few remarks.

The amount of resources in terms of transmission links in the sub network is more efficient used in a packet switching network than in a circuit switching network, if the traffic flow between users is very bursty. Take for instance a user who edits a program on a remote computer. Measurements on this kind of interactive use of computers show that the sum of the time periods when communication resources are actually needed is seldom more than 5% of the total set up time.

The transmission error detection and correction mechanisms are usually more easily and more powerfully implemented in a packet switching network. Errors caused by noise on the transmission links are handled by error check bits in the packet header and if the detected errors can not be corrected, a packet retransmission is done from the previous node. In [52] Kleinrock says: "Tests have indicated that the raw phone line packet error rates vary from about one in 10^3 to one in 10^5 ; these often bursty errors appear not to affect network performance and, as far as we know, no undetected errors have as yet passed through the ARPANET".

In a packet switching network a mismatch between the

sender and the receiver regarding the ability of sending and receiving data at a certain speed is overcome, for the users by means of the store-and-forward principle and for the sub network by means of different end-to-end protocols.

It is obvious that the circuit switching principle must have advantages in applications where the users send large amounts of data at a time, for instance long file transfers. Once a connection is set up, the sub network resources, in terms of transmission capacities within the established connection, could be used solely by the data transferred between the two users, and sub network resources, in terms of switching computers, are available to other users.

Moreover, a user who has been allocated a path in a circuit switching network could be (almost) sure to have his physical connection established as long as he will or can afford.

With these remarks we now focus our discussion on packet switching networks and principles.

2.4 Protocols

Protocols could be seen as a set of agreements on the format and relative timing of information to be exchanged between two processes that communicate over some communication link [74].

In packet switching computer communication networks protocols are conveniently partitioned according to implementation levels in the network as well as to the effects that should be obtained by the protocols.

Protocols in a network can usually be seen layered and one speaks of high or low protocols. The lowest level protocols involve reliable packet exchange between nodes, see figure 2.2.

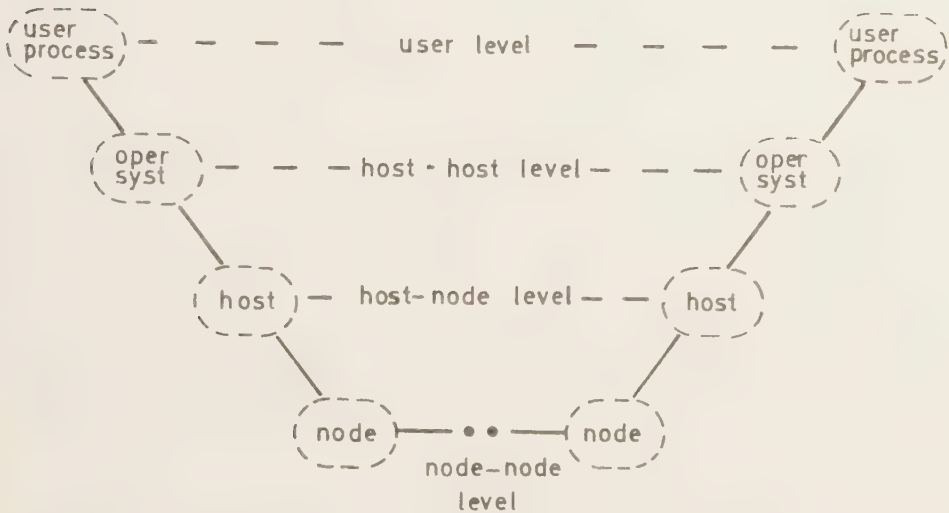


Figure 2.2

Protocols on this level usually involve transmission error detection and correction, flow control to avoid packet congestion and routing. (Flow control is discussed in the next chapter and different routing policies will be discussed later in this section.) The next level handles the information exchange between the host computers and the nodes. On this level protocols put constraints on the host computers regarding the amount of data that is to be sent. Protocols on the host to host level construct and maintain communication between processes in two host computers. A pro-

cess that requires communications with another in some remote computer system has to make a request to its own host computer to establish and maintain the communication under the host to host protocol. Finally we find the high level protocols on user level, i e a protocol between two user processes. On this level, users are provided primitives for the establishment of a connection, the transmission of data etc.

Besides the four levels mentioned we usually find a protocol for communication between the origin and the destination nodes.

The second way of structuring protocols is by means of their specific tasks. The three major tasks for protocols are error handling, flow or congestion control and routing.

A packet switching routing procedure is a set of rules that determines the next node a packet will visit on its way through the network [25]. In a network that uses a fix routing procedure, the choice is based solely on the origin and the destination node. Thus when a packet enters the network its way through the network is predestined and fixed. In order to decrease the effects of transmission link and node failures many networks use an alternative routing procedure, i e if the chosen "next node" is unavailable or the corresponding transmission link is not usable the routing procedure makes a second choice. An adaptive routing procedure makes a decision based on both the origin and the destination of the packet and the current load situation in the network. Thus packets could be routed away from a congested part of the network. This last routing procedure requires a continuous updating of each node of the traf-

fic situation in the entire network. This is normally done by routing information sent between all adjacent nodes. This in turn causes an overhead traffic in terms of routing information and the fact that packets sometimes are routed a longer way through the network [55].

2.5 Different sub network topologies

The topology of the interconnections in a network is of great concern since it has a major effect on the performance of the distributed system. In this section we will discuss different possibilities for the topology of the sub network [28].

In a sub network with just one node, a node to which all the host computers are connected (see figure 2.3),

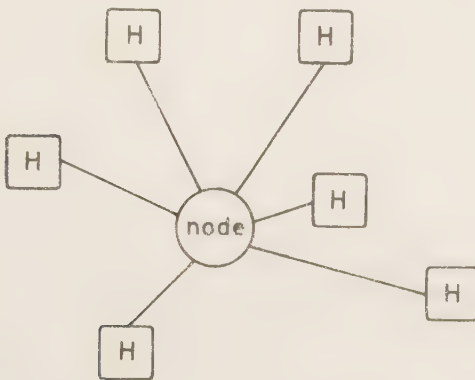
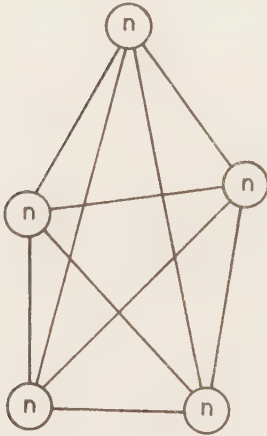


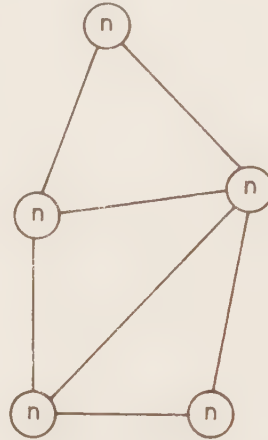
Figure 2.3

the reliability of the node is very crucial for the reliability of the entire network. As soon as the central node exhibits faulty functions the entire system breaks down. The advantages of such a system are low transit delays and low complexity.

In a meshed network the effects of a node or a link failure can, for a major part of the network users, be overcome by means of an alternate or adaptive routing procedure (see figure 2.4).



fully connected
network



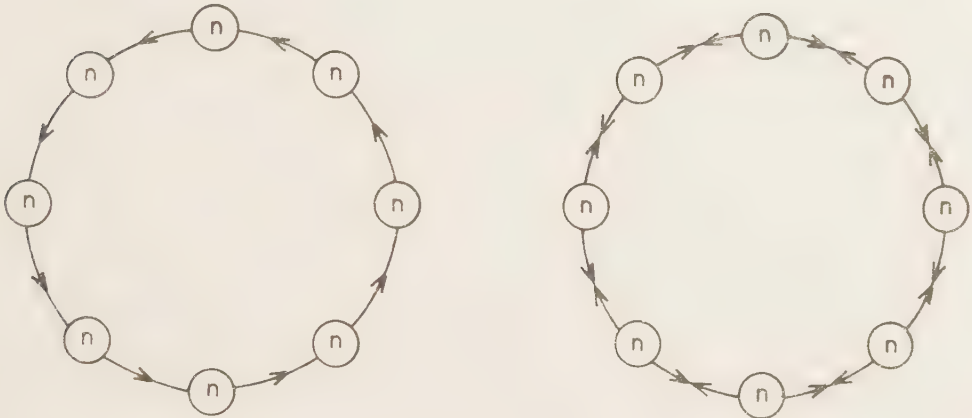
distributed network

Figure 2.4

Within the group of meshed networks we find both fully connected networks, distributed networks and loop or ring networks. A fully connected network, in which every node is connected to every other node directly, is usually only justifiable in a network with a few nodes.

A loop network, which is found especially in small local networks, can either be single or bi-directional (see figure 2.5). The advantage of the bi-directional loop network lies in the possibility of having two paths between any two nodes, whereby a single transmission failure does not affect the communication between any

two nodes but merely an increase of the transmission delays.



loop or ring networks

Figure 2.5

2.6 A brief survey of existing packet switching networks

At the National Physical Laboratory in the U K the Division of Computer Science has conducted research into the principles of packet switching networks since 1966. These efforts brought about the construction of the NPL data communication network in 1968 [84]. The network employs the store-and-forward packet switching method of operation though the sub network consists of just one node.

The wellknown ARPANET (The US Department of Defence Advanced Research Projects Agency Network) was built in 1969 and developed fast from a four node network in 1969 to a 34 node network in september 1972 [38],[52]. The

network provides a message switching service and is geographically distributed across the continental US and extended by satellite to Hawaii and to a few nodes in Europe. The ARPA network is a private network and is not intended to compete with comparable commercial service. Most of the transmission links in the sub network, connecting about 60 nodes, are leased 50 kbits/sec fully duplex synchronous channels and the host to node connections are 100 kbits/sec asynchronous channels. The topology of the distributed network provides at least two physically separate paths between each pair of nodes. The network is a message switched store-and-forward network to which the host computers deliver messages of maximum 8063 bits. These messages are divided into packets in the sub network where each packet contains at most 1008 bits. Each packet of a message is transmitted independently to the destination node, which reassembles the message before shipment to the destination host. After the reassembly an end-to-end acknowledgement is sent back to the source host.

The network uses an advanced adaptive routing policy, i e the choice of the next node is based on both the destination of the packet and the current traffic situation in the network.

The dominant type of errors, i e errors caused by noise on the transmission links, is handled by error detection and correction and also by retransmission. This procedure is performed between each pair of adjacent nodes.

The network was designed to achieve both a rapid delivery for the short interactive messages as well as a high throughput for long file transfers. If we exclude transmissions over satellite links, which introduce a

delay of approximately 250 ms just due to propagation, the first of the two goals is fulfilled. The typical response times for short messages vary between 100 and 250 ms. However, as regards the second goal, the network is able to provide a high bandwidth for long file transfers under moderate loads, but fails somewhat under heavy traffic loads.

The CYCLADES network in France is a non public packet switching network linking research centers and universities. It has been in full operation since 1974 [72], [90],[35],[73]. The sub network, called CIGALE, provides its users with a datagram service, i.e. the packets within a message are transmitted independently of each other. The present topology is shown in figure 2.6.



The CYCLADE network

Figure 2.6

If we regard the CIGALE network as an inner layer of the total network, the next layer is the interprocess communication facility called the Transport Service. This service is provided by Transport Stations located in each data processing equipment and cooperating through CIGALE according to an end-to-end transport protocol. The higher layers correspond to application oriented protocols.

In the network an adaptive routing policy is used. Furthermore packets are dropped under exceptional circumstances, i e when congestion control is defaulted.

The Datapac network in Canada [18],[87] is a nationwide public packet switched data communication network operated by the Trans-Canada Telephone System. The commercial service began in 1977 with five installed nodes and since then Datapac has grown to a size of 14 installed nodes and 24 transmission links connecting these nodes. The communication facilities provided by the nodes are implemented using three logical layers. At the core, the datagram layer provides a basic inter-nodal communication facility. On top of that layer a virtual circuit communication layer is built to provide the basic Datapac virtual circuit service. Customer access to the virtual circuit service is provided by the network access layer, which in Datapac includes the CCITT interface X.25 for synchronous lines.

Datagrams within the sub network are treated as distinct autonomous units of information. They are routed and transmitted through the network on the basis of the destination node identification, which is contained in the header of each sub network packet.

The sub network may be characterized as a lossy medium,

i e the sub network discards packets under congested situations.

Furthermore duplicate packets may be generated by re-transmissions on a different trunk after a trunk or node failure. Thus two copies of a datagram could possibly arrive at the destination node. These failures are handled and corrected by the virtual circuit layer.

The list of existing packet switching networks today could be very long. Let us for short just mention some of the other major public data networks in Europe and Northern America.

Transpac is the French public data network [24]. Tele-net [78], a huge ARPANET-like public network in the US with almost 100 nodes spread over the US continent. The British PSS network (Packet Switched Service)[37],[2], with six nodes in London, Manchester and Glasgow. TYMNET [89] in Canada, which is a commercial value added network and which has been in operation since 1971.

2.7 Interconnection of networks

It is a natural user requirement that a terminal or local computer should have the ability to access any computing resource desired, even if the resource is connected to some other computer communication network.

While the technology of individual networks is rather advanced, with a great number of networks already in operation, the questions, issues and problems of network interconnection are, though a number of networks have been connected successfully over the last few years,

neither answered nor solved. Network interconnections raise a great many technical and political questions and issues.

The term "gateway" has been adopted in many articles [15],[88],[14],[33] to stand for the connection between the involved networks. The gateway, however, need not imply a physical device which joins a pair of networks, but may merely be software in a pair of nodes in different networks.

The number of technical issues that arises when networks are to be interconnected contains of course flow and congestion control, routing and so on. Depending on how the gateway is implemented in an interconnection, the magnitude of these problems varies.

If the gateway is implemented at the switching node level, i e if the gateway could be seen as a node in both networks, it has to map the entire node to node protocol of one network into those of others. This often complex and unflexible method usually requires the involved networks to be very similar.

A parallel connection at host level, which is more flexible, requires that the gateway contains a complete internal host for each attached network.

Gateways could also be imolemented as a series connection at host level.

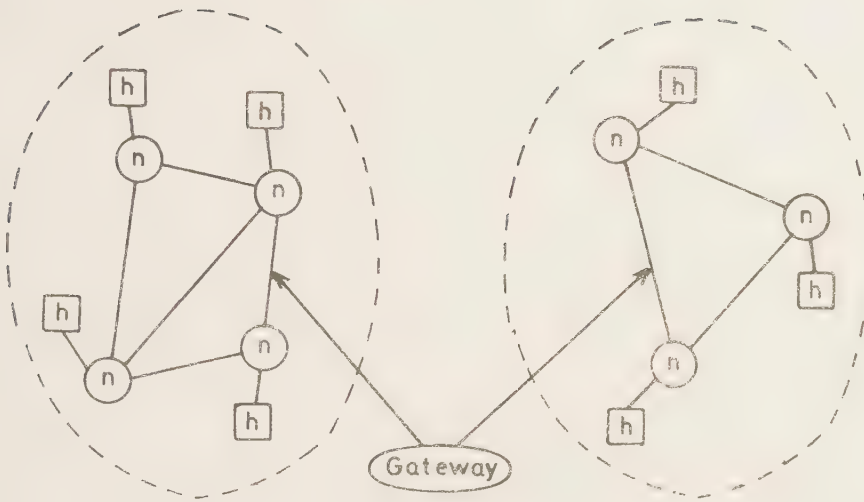


Figure 2.7 Gateway on the switching node level

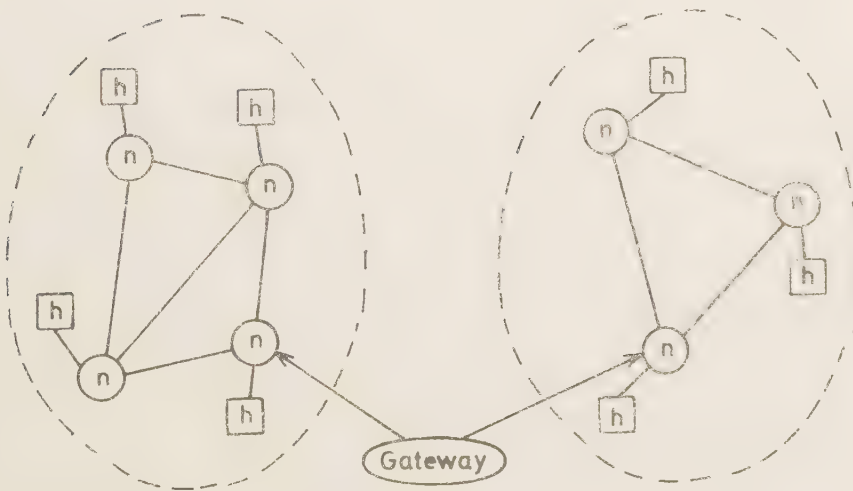


Figure 2.8 Parallel connection at host level

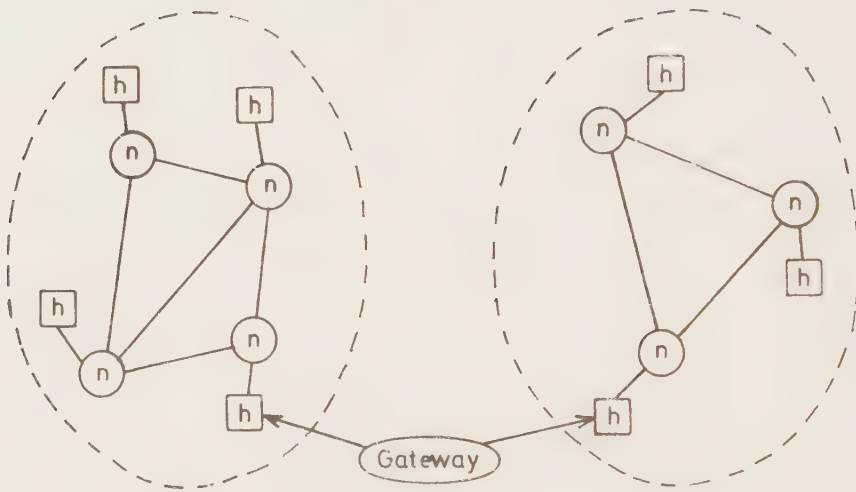


Figure 2.9 Series connection at host level

Though an important work of standardization has been made, that resulted in the X.75 interconnecting networks protocol, many questions are still not answered. The coming proliferation of computer networks demands prompt attention to the problem of interconnecting heterogeneous networks if the full potential of computer communication is to be realized.

2.8 Computer networks in the eighties

The issue of interconnecting networks will still during the eighties be a major task. The discussions in the previous section concerned primarily packet switching networks. However, the interconnection must also be done between packet and circuit switching networks.

One change which will occur in computer communication networks is the incorporation of broadcast satellite facilities.

Another major change will be the introduction of packet radio. Since local distribution is by far the most expensive portion of a communication network, ground radio techniques are of considerable interest as they can be used for local distribution.

The basic concept in packet radio is to share one channel among many host computers or terminals, each of which only transmits in short bursts when it has real data to send.

Packet radio will emerge as a viable technology for both fixed and mobile computer communications. Today, however, the costs for the transmitter and the receiver are in most cases too high to motivate a wider use.

The economic advantage of dynamic allocation, like packet switching, over preallocation, like circuit switching, will soon motivate the former principle to be used in not only computer communications application. Take for instance our conventional telephone communication system. Digitized voice can be essentially compressed by packet switching since in normal conversation each speaker is only speaking about one third of the time[20].

Teletext and videotext are other examples of communication forms suitable to be implemented by means of packet switching.

This will lead to an integration of all kinds of communication. Any kind of information, whether it is voice, video or data, will be transmitted in the same network,

not necessarily a packet switching network, but anyhow a network which uses dynamic allocation.

CHAPTER 3 - QUEUEING AND QUEUEING NETWORK THEORIES

3.1 Queueing theory - a unique tool for understanding the behaviour of complex communication systems

In his book Queueing Systems, volume 2, Computer Applications [52], Kleinrock expresses the potentials within queueing theory as:

In fact, the use of queueing theory to analyse resource allocation and job flow through computer systems is perhaps the only method available to computer scientists in understanding the behaviour of the complex interconnection of their systems.

We would like to extend this statement to incorporate not only computer systems, but furthermore the great area of communication systems. Moreover, the concept "communication systems" should be seen in a very wide sense. The word "communication systems" lend many non-specialists to think of at first hand a telephone communication system and perhaps sublease of a telex communication system.

However, within the conception "communication systems" we also find different kinds of communication systems all the way from two coupled computers to nation wide and even world wide systems. Furthermore, the computerized control system of many complex technical systems like an SPC(Stored Program Control) telephone exchange, do within themselves incorporate a communication system; i e there exists a complex communication between the sometimes large number of control computers. Even within a single computer system, there exists vast communication between CPU, main memory, disks, tapes, terminals etc.

The queueing models will not only give us information on the modelled system in terms of for instance delays in and utilization of the system, the number of customers finding the system filled due to buffer limitations etc. They are also useful during the planning of a new system. Questions like: how big a capacity do we need for the central processor in an SPC exchange; how many PCM(Pulse Code Modulation) links do we need between two SPC exchanges; what channel capacity between a CPU and a disk system do we require, do often find their answers by means of queueing theory.

At last, but not of least importance, our queueing models will help us to understand how complex communication systems are built up and the reasons for the chosen system solution. (Well, let us assume that there are good reasons.)

3.2 Definitions

Let us start to look at a server system in order to define and describe the often unpredictable behaviour of the customers [19],[22]. Depending on the application, customers could be anything from packets in a computer communication system to programs in a traditional computer system. In our server systems we describe the arrival streams of the customers in terms of a stochastic process, i e the arrival process is described in terms of the probability distribution of the interarrival times of customers and is denoted $A(t)$, where

$$A(t) = P[\text{time between arrivals} \leq t]$$

Furthermore we usually require the arrival process to be a stationary renewal process, i e the interarrival times are independent and identically distributed random variables.

The customer behaviour is also described by their need for service and we use the notation $B(x)$ for the appropriate probability distribution

$$B(x) = P[\text{service time} \leq x]$$

A model of a queueing system is usually described by the wellknown three concepts notation

$$A/B/m$$

where A describes the arrival process of the customers, B the service distribution and m the number of servers. Thus A and B could be taken from for instance the set M, D, E_r , H_r and G, where M stands for Markov, i.e. the exponential distribution, D for Deterministic, E_r for Erlang-r, H_r for Hyper-r exponential and G for General. Furthermore the three concepts notation implicates that the server system has unlimited waiting room, customers are taken from the queue to the server(s) on a First Come First Served (FCFS) basis etc. If we want to use a model with for instance N queue places or a LCFS (Last Come First Served) queueing discipline, this is indicated after the three concepts notation

$$M/D/1 \cdot N \text{ queue places}$$

$$M/M/2 \cdot LCFS$$

The last notation describes a two server system with an unlimited number of queue places, customers are transferred from the queue to one of the servers on a LCFS basis. Furthermore

$$A(t) = 1 - e^{-\lambda t} \quad t \geq 0 \quad (3.1)$$

where

$\frac{1}{\lambda}$ is the mean interarrival time

$$B(x) = 1 - e^{-\mu x} \quad x \geq 0 \quad (3.2)$$

where

$\frac{1}{\mu}$ is the mean service time.

For some of these queueing systems we can derive the steady state probability density function

$$p_k = P(k \text{ customers in the total system})$$

For others we are just able to derive the corresponding z-transform, i e

$$Q(z) = E\{z^k\} = \sum_k z^k \cdot p_k \quad (3.3)$$

By means of the steady state probability distribution or the z-transform we are able to derive the mean number of customers in the system $\bar{N}$.

$$\bar{N} = \sum_k k \cdot p_k \quad (3.4)$$

$$\bar{N} = \left. \frac{\partial Q(z)}{\partial z} \right|_{z=1} \quad (3.5)$$

In some cases we can even derive the waiting time distribution

$$W(y) = P(\text{waiting times} \leq y)$$

Given the waiting time distribution, it is of course possible to derive the mean waiting time

$$W = \int_Y y \partial W(y) \quad (3.6)$$

However, Little's result [64] gives a relation between the mean number of customers in the system, mean time in system T and throughput λ .

$$\bar{N} = \lambda T \quad (3.7)$$

and since

$$T = \frac{1}{\mu} + W \quad (3.8)$$

we also have

$$W = \frac{\bar{N}}{\lambda} - \frac{1}{\mu} \quad (3.9)$$

Little's formula is very useful in various systems including those with a limited number of queue places. However, one has to be extremely careful when using it. In a system with a limited number of queue places we have to define the mean waiting time carefully. For a network with just one server system it is obvious how the mean waiting time should be defined: the expected value for the non blocked customers. However, in a network model, which incorporates a number of server systems, this definition is not that obvious (see chapter 10). Furthermore in a system with a limited number of queue places the effective arrival rate λ_e should be used in the formulae.

3.3 The M/M/1 system

In the early packet switching network analyses, the individual transmission links and the appropriate buffer space in the node were modelled as an M/M/1 system. Thus the buffer limitations in the nodes were not in-

cluded in the models.

For the M/M/1 system we assume

$$A(t) = 1 - e^{-\lambda t} \quad t \geq 0 \quad (3.10)$$

and

$$B(x) = 1 - e^{-\mu x} \quad x \geq 0 \quad (3.11)$$

The wellknown steady state distribution is given by

$$p_k = \left(\frac{\lambda}{\mu}\right)^k \left(1 - \frac{\lambda}{\mu}\right) \quad (3.12)$$

or if we for short let $\frac{\lambda}{\mu} = \rho$

$$p_k = \rho^k (1 - \rho) \quad (3.13)$$

The mean number of customers in the system

$$\bar{N} = \frac{\rho}{1 - \rho} \quad (3.14)$$

and mean waiting time

$$W = \frac{1}{\mu} \frac{\rho}{1 - \rho} \quad (3.15)$$

Furthermore, the mean time spent in the system

$$T = \frac{1}{\mu} \frac{1}{1 - \rho} \quad (3.16)$$

These performance measures are of course related

$$\bar{N} = \lambda T \quad (3.17)$$

$$T = \frac{1}{\mu} + W \quad (3.18)$$

In figure 3.1 these performance measures are shown as a function of load.

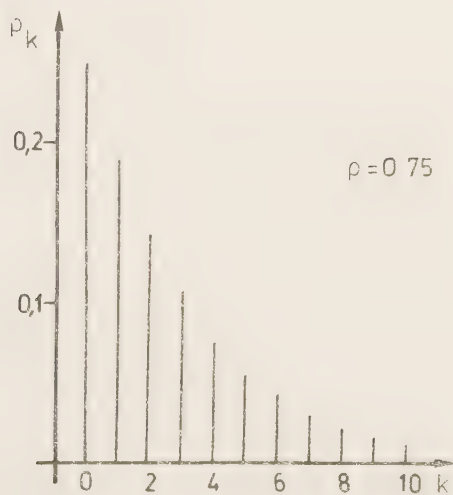
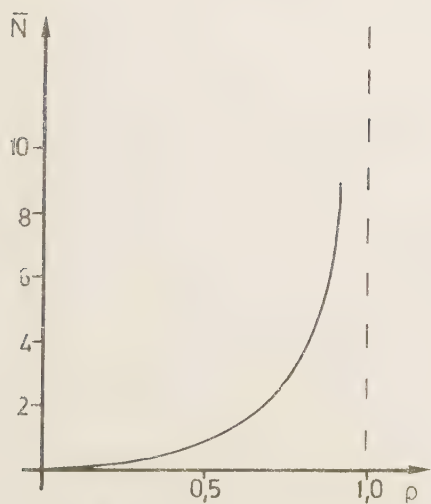
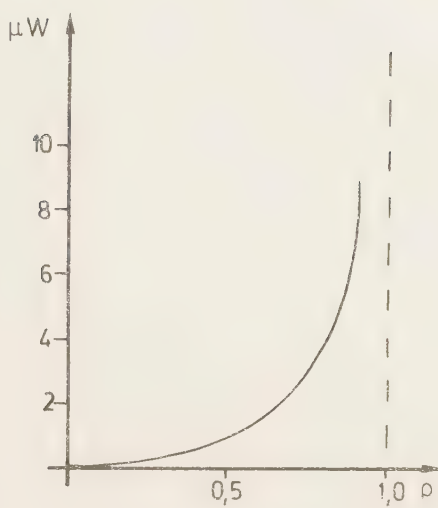
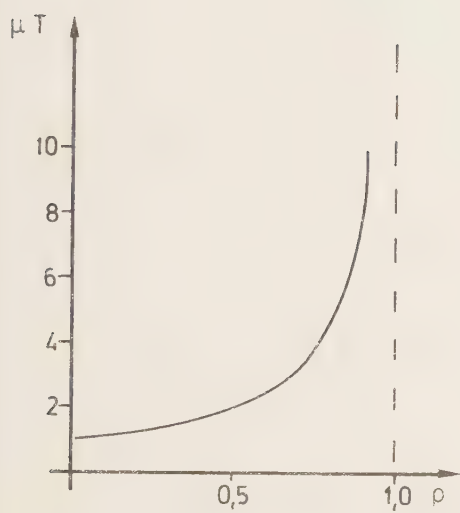


Figure 3.1

For a single server system with no limitations, like for instance the number of queue places, we require $\lambda < \mu$ and thus $\rho < 1$. In figure 3.1 we find the number of customers, the mean time spent in the system and the mean waiting time increasing dramatically as ρ approaches unity. This effect is caused by the variability in both interarrival time and the service time.

Furthermore, for our M/M/1 system, as in fact for the G/G/1 system, $p_0 = 1 - \rho$ and thus the probability of finding our system busy is ρ .

However, for the M/M/1 · N queue places, as for any system with this limitation, we do not require $\lambda < \mu$ to obtain a stable system. Here the steady state probability distribution is given by

$$p_k = \rho^k \frac{1 - \rho}{1 - \rho^{N+2}} \quad (3.19)$$

The mean number of customers in the system

$$\bar{N} = \sum_{k=0}^{N+1} k \cdot p_k \quad (3.20)$$

$$\bar{N} = \frac{(\rho - (N+2)\rho^{N+2} + (N+1)\rho^{N+3})}{(1-\rho)(1-\rho^{N+2})} \quad (3.21)$$

Here T is given by Little's formulae

$$T = \frac{\bar{N}}{\lambda_e} \quad (3.22)$$

where

$$\lambda_e = \lambda(1 - p_{N+1}) \quad (3.23)$$

⇒

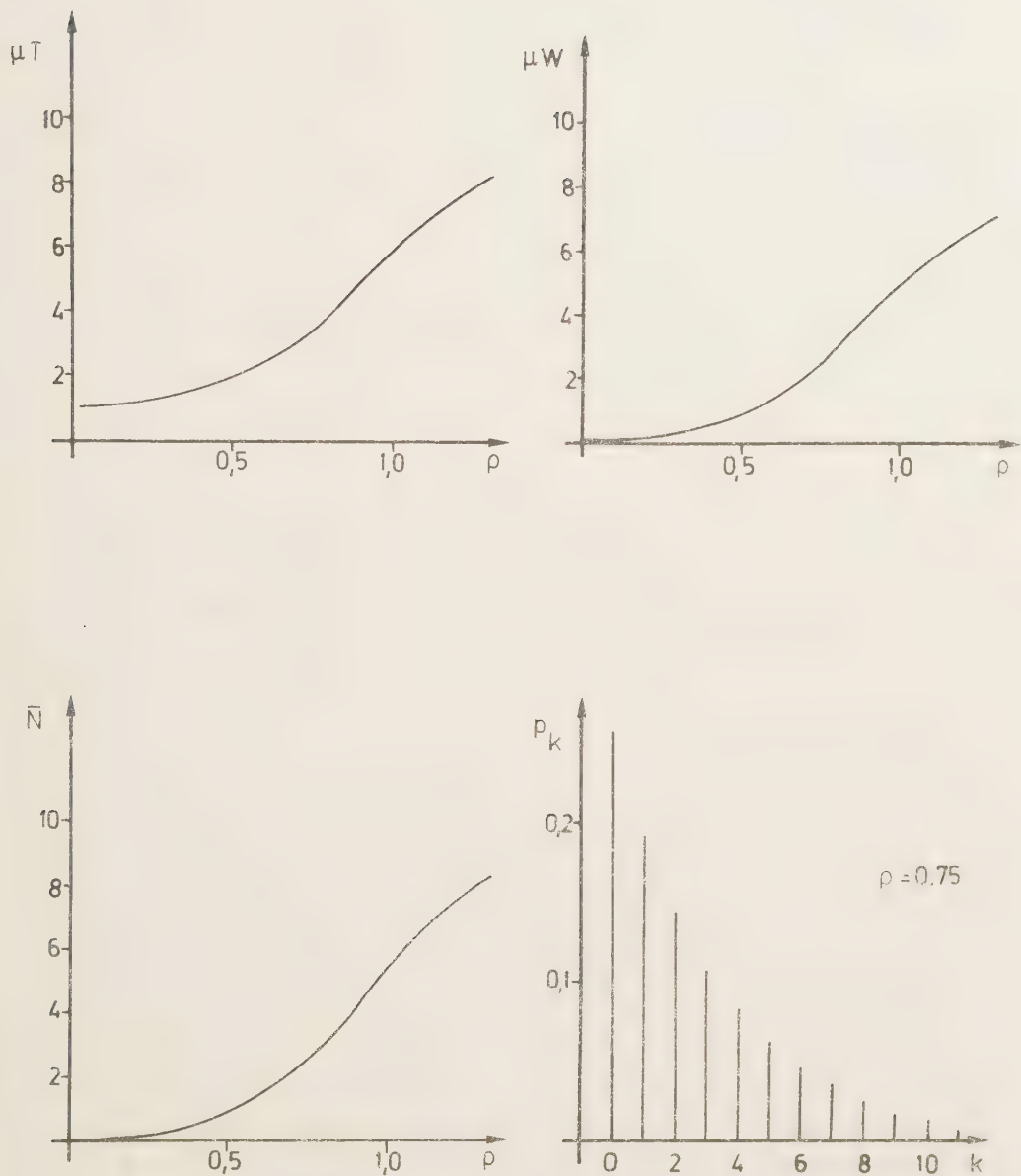


Figure 3.2 $N = 10$

$$T = \frac{1}{\mu} + \frac{1}{\mu} \frac{\rho}{1-\rho} - \frac{1}{\mu} \frac{(N+1)\rho^{N+1}}{(1-\rho^{N+1})} \quad (3.24)$$

and as earlier

$$W = T - \frac{1}{\mu} \quad (3.25)$$

Moreover the probability that a customer arrives to a system that is filled, i.e. the blocking probability, is of course given by

$$p_{N+1} = \rho^{N+1} \frac{1-\rho}{1-\rho^{N+2}} \quad (3.26)$$

These performance measures are drawn in figure 3.2. Here we find a completely different behaviour in comparison with the unlimited case.

3.4 The G/G/1 system

When we talk about the G/G/1 system we usually assume the arrival process and the service distribution being renewal processes [21]. Though having excellent works of Kingman [50], Andersson [3] and Beněš [6], we are now dealing with a system for which not even the mean waiting time is known.

However, if we use the method of stages, we are able to form solutions for systems with a spread spectrum of the coefficient of variation both for the arrival and the service processes. The solutions are generally given in terms of transformations impossible to solve analytically.

If we restrict one of the processes to be Markovian, i.e.

either an M/G/1 or a G/M/1 system, solutions of the steady state probability distribution and the waiting time distribution are given as transforms, i.e. $P(z) = \sum z^k \cdot p_k$ and $W^*(s) = \int_0^\infty e^{-st} w(t) dt$. The average values, however, are given explicitly. Take for instance the wellknown Pollaczek-Khinchin formula for the M/G/1 system

$$\bar{N} = \rho + \frac{\lambda^2 \overline{x^2}}{2(1-\rho)} \quad (3.27)$$

where $\overline{x^2}$ is the second moment of the service time distribution and

$$W = \frac{\lambda \overline{x^2}}{2(1-\rho)} \quad (3.28)$$

is given by means of Little's formula.

Even for the M/G/1 + finite waitingroom, solutions regarding the basic performance measures are known [76], [56], [62].

3.5 Queueing Network Theory

Queueing network theory could be seen as an enlargement of queueing theory modelling systems in which a customer requires service at more than one station or node. Thus we may think of a network of nodes, each of which is a service center, perhaps with multiple servers at some of the nodes and each with storage room for queues to form.

Within queueing network theory we assume customers to enter the network at various nodes, especially the external arrival rate at node i is generally assumed to be Poisson with rate $\gamma_i \forall i = 1, 2, \dots, N$; (we assume there are N nodes in the net). When a customer has gained service at node i , perhaps after spending a certain time in the

appropriate queue, he is with probability r_{ij} transferred to node j or leaves the network with probability $1 - \sum_{j=1}^N r_{ij}$. Jackson [43] addressed himself to this problem and assumed the individual nodes to be primarily M/M/m systems with a FCFS queueing discipline. In section 3.6 we will further find that the departure process from such a system is Poisson.

The state variable for this system of queues could be seen as a vector $\underline{k} = (k_1, k_2, \dots, k_N)$, where k_i is the number of customers in the i :th node. Jackson showed that the state probability density function for this system is given as a product form solution

$$p(k_1, k_2, \dots, k_N) = p_1(k_1)p_2(k_2) \cdot \dots \cdot p_N(k_N) \quad (3.29)$$

i.e. the solution is given as the product of the steady state probability density functions of the individual M/M/m systems.

Gordon and Newell [34] extended the works of Jackson and gave results concerning closed queueing networks, i.e. networks with a finite and fixed number of customers, a network where $\sum_{j=1}^N r_{ij} = 1$ and $\gamma_i = 0 \forall i$. Again the state probability density function formed a product form solution, in this case, however, normalized with a constant. Basket, Chandy, Muntz and Palacios [5] extended the results to include different queueing disciplines, multiple classes of jobs and nonexponential service time distribution.

Models, based on the multiclass queueing network theory with possibilities of letting the network be opened, closed, limited or mixed, seem to have the potential of being credible. Extensive validations

within various fields have verified that these models reproduce observed performance measures with remarkable accuracy [13],[32],[39],[63],[80].

However, though much effort has been made regarding efficient computational algorithms [12],[11] for multi-class queueing networks, the amount of variables and the huge state space do sometimes make the models intractable.

There is little doubt that much of the future research in queueing networks will be directed toward approximate methods and more general applications [68]. The underlying reasons for this are of course the answer to the question whether our models are credible or not. Perhaps we have to make them less credible in order to gain exact solutions. However, within queueing network theory we have to develop tools in order to form credible models, which perhaps have to be solved approximately. In [16] the conjecture of Chandy and Sauer is that queueing network models with credible assumptions can be solved approximately to provide credible performance estimates at low cost.

3.6 The exponential assumption

In order to strengthen our conviction of the credibility of our coming models, we have to have a discussion regarding our exponential assumption that will be made.

Here the discussion will be concentrated to the arrival and the departure processes. A brief discussion regarding the credibility of exponential service times will be held in chapter 6.

As we for our network models will assume that packets with origin node s and destined for node t will be generated (of course in node s) according to the Poisson process, and as we know that random merging and splitting of independent Poisson processes will form new Poisson processes, it is important to know whether or not the departure process from our node models are Poisson or not.

Results due to Burke [9] say that the steady state output of a stable $M/M/m$ queue with input parameter λ and service time parameter μ for each of the m channels is in fact a Poisson process at the same rate λ . Burke also established that the output process was independent of the other processes in the system.

As we discussed when we described the notation $A/B/\cdot$, it is understood when claiming something for an $M/M/m$ system that the queueing discipline is FCFS. In fact the $M/M/m$ system is the only such FCFS system with this property.

We call this property the "Markov implies Markov property" and it holds for the following most important systems

- i) the $M/M/m$ system
and though FCFS is understood within the notation it is worthwhile a note
- ii) the $M/G/1 \cdot$ processor sharing
i.e. a Round-Robin queueing discipline where the service quantas are reaching zero asymptotically
- iii) the $M/G/1$ LCFS-PR
PR: Preemptive-Resume
i.e. the last arriving customer does immediately take the server and the customer thrown out from the server is placed first in the queue.

iv) the $M/G/\infty$

i.e. the infinite server model.

Well, one thing is certain, our $M/M/1$ queue with limited restricted buffer sharing policies will not have a Markovian departure process.

However, the known results concerning merging and splitting renewal processes [23] do talk in favour of our assumptions. The merging and splitting of processes in a meshed network will make the coefficient of variation and higher moments approach those of the Poisson process.

Take for instance the random splitting of a primary renewal process into n secondary processes. A renewal in the primary process, i.e. the one we split, is considered to belong to the j :th secondary process with probability α_j ($\sum_{j=1}^n \alpha_j = 1$). If the primary process has the coefficient of variation C_X^2 the one corresponding to process j will be

$$C_{X_j}^2 = ((C_X^2 - 1)\alpha_j + 1) \quad (3.30)$$

which tends to unity as $\alpha_j \rightarrow 0$.

Another crucial point regarding the credibility of our coming models is the frequently used independent assumption. Let us describe this assumption with an easy example, the two stage $M/M/1$ tandem network. In our model,



Figure 3.3

when a customer has entered system 1 or the server at system 1, a service time is considered drawn from the exponential distribution. However, when the same customer enters system 2, a new independent service time is drawn. If this model is supposed to describe the flow of packets, the length of the packet should be the same in the two systems and thus, even if the two systems have different service capacities, the two service times should be equal or just differ by a constant.

Much effort has been made in clearing out to what extent this assumption contributes to non-credible models. Contrary to pure or almost pure tandem models the effects within meshed networks are marginal [53],[29].

CHAPTER 4 - FLOW CONTROL - CONGESTION CONTROL

4.1 Introduction

In various real time systems, like a communication system, there exist mechanisms to prevent an overload demand for service.

Within the field of computer communication networks the need for such mechanisms are much more accentuated for packet switching networks than for circuit switching networks. In a circuit switching network, once network resources are offered to a user, the user is guaranteed the use of these resources until completion of the communication. As opposed to this many packet switching networks are built based on the knowledge that the need for communication for many users are of a very interactive form. This usually leads to that network users will only occasionally require that the virtual connection set up by a packet switching network is converted into a real connection for the transmission of data. Should, however, user demands in a packet switching network be allowed to exceed the system capacity, unpleasant congestion effects occur, which rapidly destroy important network performance characteristics and efficiency advantages.

4.2 Negative effects in uncontrolled packet switching networks

The negative effects could be observed in terms of degradation of throughput, unacceptable delays and deadlock situations.

The relation between throughput and offered load will

for many of the uncontrolled dynamic systems be of the kind that is shown in figure 4.1.



Figure 4.1

Deadlock situations could occur as an effect of various situations. In packet switching networks one obvious situation is the one indicated in figure 4.2.



Figure 4.2

Packets in node A routed towards node B have to wait due to buffer limitations in node B and the same applies to packets in node B routed towards node A.

Another locked situation is the one corresponding to packet reassembly in a destination node. We put our attention to a part of a network (see figure 4.3) in which node C has to have the five packets P_1 , P_2 , P_3 , P_4 and P_5 to make a packet reassembly and after which node C can deliver the appropriate message to the connected node. (P-packets as well as M- and N-packets are bound for node C.)

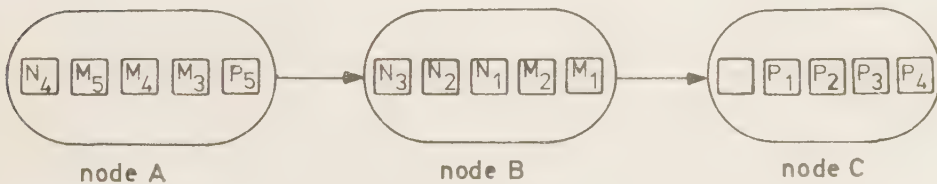


Figure 4.3

Deadlock situations could also occur as a consequence of loop congestion in a network. Node A waits for buffers in node B to become free, node B waits for buffer space in node C, which in turn waits for space in node A.

Network congestion could also be caused by a mismatch between the sending and the receiving host. If the receiving host is much slower than the sending host, the node connected to the receiving host will be congested and this congestion will rapidly spread to the complete network. This mismatch could also often lead to a blocking of the receiving host.

Another cause of congestion is the interaction between transit traffic and new traffic. As transit traffic already has been provided with resources from the network, the gained resource should be wasted if a transit packet was lost.

The main cause of throughput degradation in a packet switching network is waste of resources caused by a congested situation. This waste concerns both communication capacity and storage capacity.

Take for instance the ALOHA network principle [1] where packet collisions cause retransmission and thus waste of communication capacity. Another example is described in figure 4.4, where packets bound for transmission link 1 have occupied the entire buffer space and

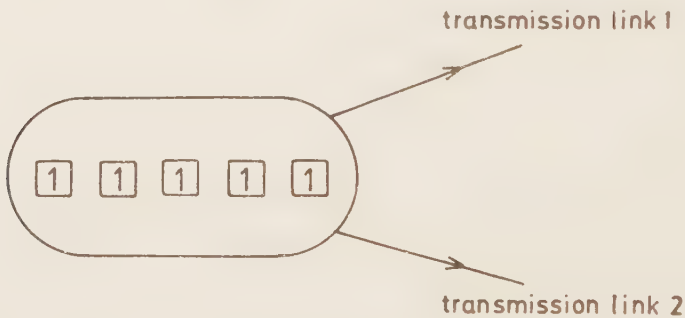


Figure 4.4

thus arriving packets bound for the non busy transmission link 2 have to be rejected due to buffer limitations.

In many articles [70] [31] the type of congestion, that occurs in an overloaded packet switching network, is compared with that of an overloaded highway system. During peak hours the demands often exceed the highway capacity, which results in a degradation of throughput

and even a total collapse of the entire system. As in packet switching networks, the merging and splitting of highway traffic and on- and off-ramp traffic cause serious negative interactions. A narrow off-ramp causes congestion on the highway network etc.

These congestion problems require mechanisms for proper control and monitoring of offered load and of transit traffic as well. The design of these mechanisms for a packet switching network is a complex task in many ways. Different mechanisms implemented for monitoring and control of a network do often interact and the interaction is usually very unpredictable.

4.3 Definitions

Many authors [81],[70],[31] have classified various control mechanisms into different groups and have also tried to isolate the two main concepts, i.e. flow control and congestion control. The lack of accepted standardized notations is of course a result of the huge number of different implementations in various networks.

H Rudin [82] uses congestion control as the overall designation for all the measures that a network can take to prevent itself from being overloaded. Flow control, then, becomes simply one of these measures. He adds, however, that the choice could have been made the other way around. But flow control is used by several authors to mean a congestion control problem, which is particular to packet switching networks. According to these definitions the three areas of congestion control are flow control, routing and scheduling.

Contrariwise Pouzin [71] defines flow control as the set

of mechanisms whereby a flow of data can be maintained within limits compatible with the amount of available resources. Flow control is then divided into two major groups, end-to-end flow control and congestion control. The former is defined as the set of mechanisms whereby an end-point receiver maintains the sender's traffic within limits compatible with the resources available at the receiver. Congestion control deals with parts of the system, for which it holds input traffic within limits compatible with the amount of resources available to the specific part of the network.

Kleinrock and Gerla [31] identify four different flow control levels in a packet switching network. Here flow control is used as the overall designation. The first level, the "hop level", restricts the traffic flow between adjacent nodes in order to avoid local buffer congestion and deadlocks. On the entry-to-exit level the object of flow control is to prevent buffer congestion at the exit node. This kind of flow control mechanism is generally implemented as a protocol between the source and the destination node. At the third level, the network access level, the objective is to throttle new traffic. This throttle is based on measurements of network congestion within the network. The fourth level, the transport level, does differ from the others, as its main purpose is to prevent congestion of user buffers at the process level, i e outside the network.

4.4 Flow control mechanisms

Which are those tools called flow control or congestion control, that could be used to monitoring external as well as internal traffic in order to avoid the variety of problems we have discussed in this chapter so far?

We start by discussing tools used on the four different flow control levels discussed by Kleinrock and Gerla. On the first level, the hop level, problems like those discussed around figures 4.2 and 4.4 arise. The problem concerning waste of communication capacity (figure 4.4) is frequently solved, at least dramatically reduced, by using restricted buffer sharing policies in the nodes. Many restricted buffer sharing policies allocate a number of buffers to packets bound for each transmission link, others state restrictions regarding the number of packets bounded for a specific transmission link, that could be stored in the buffer system.

Many deadlock problems that arise under the hop level could by a proper routing policy be delayed, though not eliminated, and thus the frequency of congestion situations is decreased.

The problems that arise under the entry-to-exit flow control level are mostly solved by means of a window scheme, that allows only up to W sequential messages to be outstanding in the network before an end-to-end acknowledgement signal is received. Furthermore, when a network congestion occurs, these end-to-end acknowledgement signals are delayed and thus they provide an extra effect in terms of delaying new traffic, corresponding to the appropriate process to process communication, to access the network. Many of the reassembly buffer deadlock problems are solved by some sort of reservation of reassembly space in the destination node.

There are a variety of mechanisms used on the network access level. Here we find the wellknown isarithmic [75] flow control scheme, which uses a limitation on the maximum number of packets that is allowed in the network

at the same time. Thus the isarithmic scheme, as well as all other schemes on this level, throttles new traffic. Another principle [46] that also throttles new traffic gives priority to transit traffic over new traffic, regarding the buffer space in the nodes.

The flow control schemes under the transport level are usually implemented within the process-to-process protocol. A frequently used mechanism is to let the receiving process send a message to the sending process, when it is ready to accept more messages.

Internal network congestion may also be reduced by rerouting some of the traffic from heavily loaded paths to underutilized paths. This should be done automatically with a good dynamic adaptive routing policy. However, as the routing procedure under a light total load should, for every packet, choose the shortest path, a rerouting must contribute to an excess of the use of link resources and thus an excess of total load. Furthermore, the routing procedure can not prevent an internal network congestion, merely delay it or reduce its frequency.

4.5 Conclusions

Though flow control mechanisms during the last five years have developed from being borrowed from "a little bay of tricks" [71] to being a little more generalized and effective, considerable investigations remain to be done. Not least the interaction of both flow control mechanisms on different levels and other mechanisms like routing [31] bring about the need for further investigations.

CHAPTER 5 - RESTRICTED BUFFER SHARING POLICIES

5.1 Introduction

During the mid sixties, when packet switching computer communication networks were modelled by among others Kleinrock [53], simplifications were made regarding the buffer space in the nodes. Though the number of available buffers was regarded unbounded, the works of Kleinrock were very important and opened up new grounds within the field of analysis of computer communication networks.

Those early network analyses gave rise to an understanding of basic performance measures in terms of transmission delays and utilization of the transmission links.

In view of the storage capacity issues, which have been observed in the ARPANET [52] and the CYCLADE net [35], the assumptions of non limited buffer space give rise to questions.

The problem of how to arrange a finite buffer space in a packet switching node has mostly been solved either by partitioning the available buffers into unique parts, where packets bound for a certain outgoing transmission link uses one specific part, or by sharing the available buffers equally between all types of packets. The former scheme, Complete Partitioning (CP), where actually no sharing is provided, leads in our models to a system of independent single servers. The latter, the Complete Sharing scheme (CS), allows any packet to use any buffer within the buffer space.

In queueing theory it is well known that, regarding transit delays, one should, when optimizing a queueing system, try to follow the "the bigger - the better" rule [51]. Thus, looking at a system with n parallel servers,

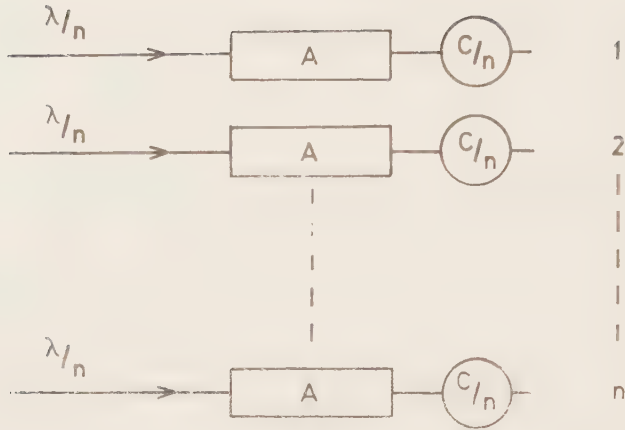


Figure 5.1 System 1

each having its own infinite queue (here we use the word parallel to mean: doing the same kind of work), one obtains a lower transit delay if the packets could share a common queue.

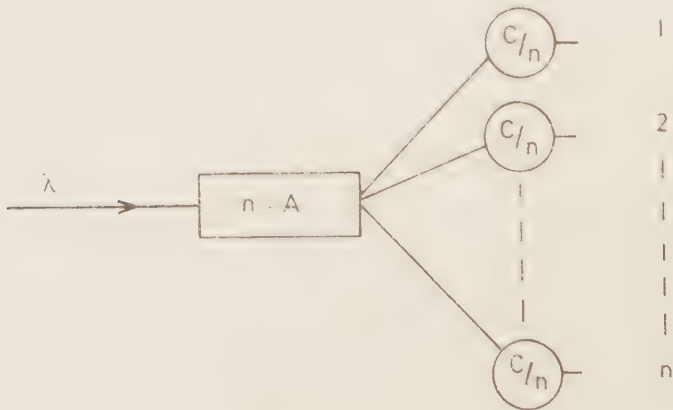


Figure 5.2 System 2

Furthermore, a system with one server, with capacity C bits per second, shows a lower transit delay than the two other systems.

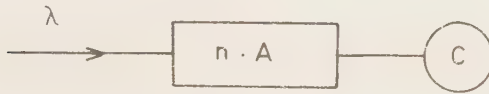


Figure 5.3 System 3

If we compare the blocking probabilities of the three systems and assume each of the $M/M/1$ systems in the first example having M/n buffers (let $M = n \cdot A$ where $A \in (1, 2, 3, \dots)$), we find the second example having a lower blocking probability than the first and the third a blocking probability lower than that of the second.

The discussion above, regarding the transit delays, assumes in the second example that every packet could use any of the n servers. However, when we compared system 1 with system 2 regarding the blocking probabilities, the result was obtained regardless of whether a packet is predestined for a specific server or not. This result ought to motivate us to use a total shared buffer space in our nodes.

However, in a packet switching computer communication network, a packet is held in one of the buffers, not only while waiting for the appropriate transmission

link (server) to become free, but also during the transmission of the packet. Thus, in spite of the transmission link being free, a packet can not be transmitted if no buffers are available.

Both the CS and CP schemes do, however, lead to undesirable behaviour of the system. Though the CS scheme yields a good system performance under low or normal traffic conditions, it fails under heavy traffic situations and in situations where load on the different transmission links are highly unbalanced. On the other hand, under the CP scheme, buffers, allocated to a slightly loaded transmission link, are hardly used at all. However, under heavy or extreme overload situations the CP scheme does very well. When we here talk about whether a scheme does behave good or not we do primarily take into consideration the blocking probabilities and the net throughput.

Under heavy load situations or when load on the outgoing transmission links is unbalanced, the performance under the CS policy is not only "not so good", but disastrous. As will be seen in the coming chapter, packets bound for the highly used transmission link or links tend to hog most, or almost all, of the available buffers and thus prevents packets bound for other transmission links to reach their links. Under heavy traffic situations, even when load on the different transmission links are equal, the CS scheme fails in securing a full utilization of all transmission links.

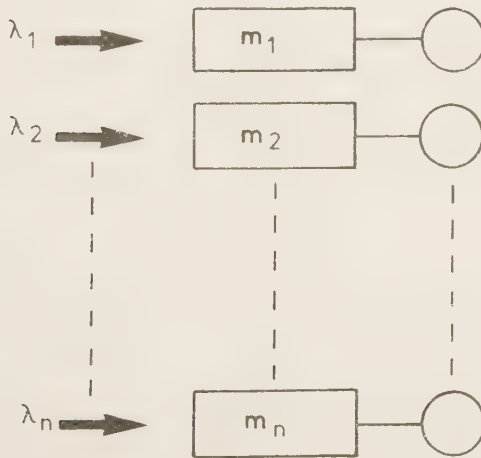
5.2 Some basic restricted buffer sharing policies

The inadequate performance of buffer arrangements using the CP and CS scheme leads to the conclusion that con-

tention for buffer space must be limited in some ways. In his Ph D thesis [24] Kamoun proposed different restricted buffer sharing policies in which the above mentioned drawbacks are reduced. Though none of them could be defined as the best, CP for instance does extremely well under heavy traffic situations, we are under certain circumstances able to judge one of them to be better than the others regarding a specific performance measure.

In order to describe the different restricted buffer sharing policies in a straight forward manner, we include the CS and the CP policies in the set.

a) Complete Partitioning CP



$$\sum_{i=1}^n m_i = M$$

Figure 5.4

Our packet switching node model can be seen as n $M/M/1 \cdot m_i$ buffers - systems. A j -packet (short for a packet bound for transmission link j) is lost if there are m_j j -packets in the system.

b) Complete Sharing CS

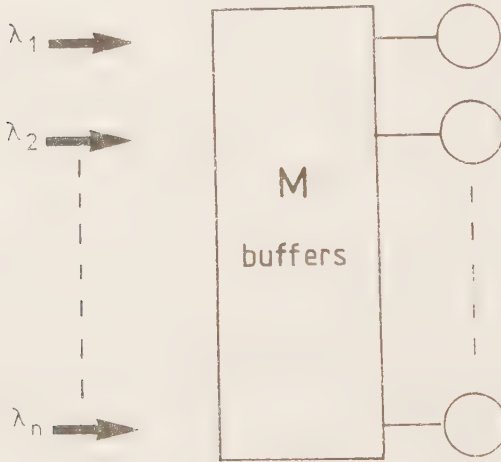
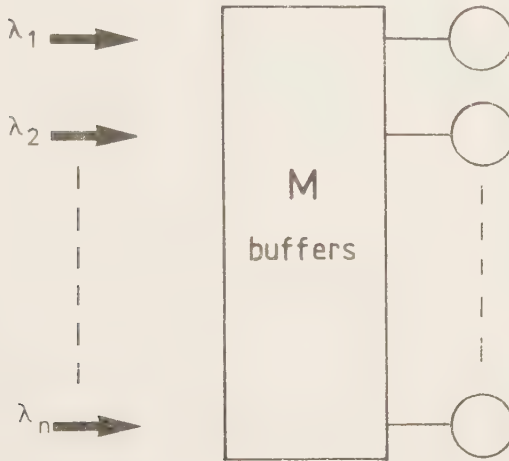


Figure 5.5

The M buffers are shared equally among the n input processes. A j -packet is lost if all the M buffers are occupied.

c) Sharing with Maximum Queues

SMXQ



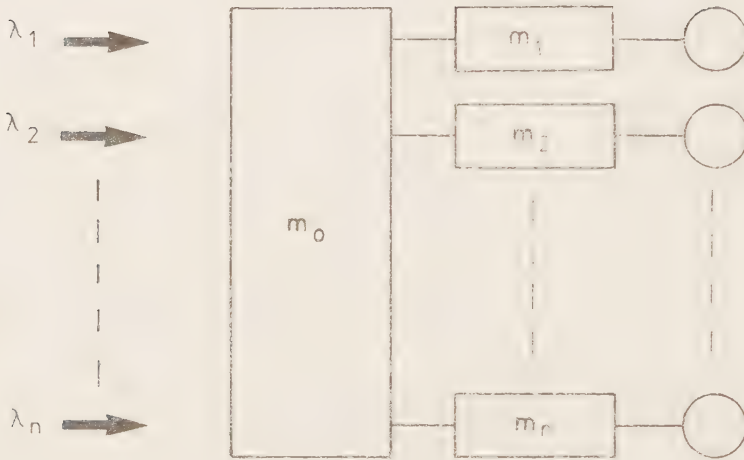
$$\sum_{j=1}^n b_j \geq M$$

Figure 5.6

Under the SMXQ policy a j -packet is lost if there are b_j j -packets in the buffer system, or if the total number of packets equals M . Furthermore we claim that

$$\sum_{j=1}^n b_j \geq M;$$

d) Sharing with Minimum Allocation. CMA



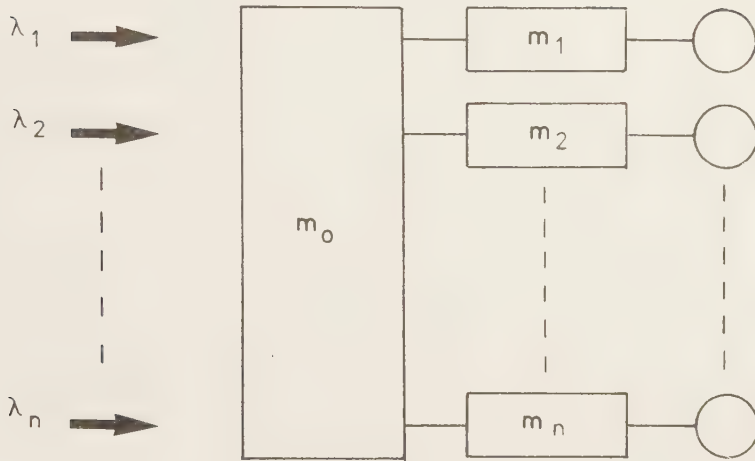
$$\sum_{i=0}^n m_i = M, \quad \sum_{i=1}^n m_i \leq M$$

Figure 5.7

A minimum number of buffers, m_j , is always allocated to j -packets and m_0 buffers are shared under a CS policy. A j -packet is lost if there are more than m_j j -packets in the system and if all the m_0 common buffers are occupied.

e) Sharing with a Maximum Queue
and a Minimum Allocation

SMQMA



$$\sum_{i=0}^n m_i = M, \quad \sum_{i=1}^n m_i \leq M, \quad \sum_{i=1}^n b_i \geq m_0$$

Figure 5.8

The SMQMA policy is a combination of the SMXQ and the SMA policies. m_j buffers are allocated to j -packets and among the m_0 shared buffers j -packets can hold up to b_j buffers. Thus j -packets are always guaranteed m_j buffers and can hold up to $m_j + b_j$ buffers.

5.3 Problems when implementing restricted buffer sharing policies in a net

When implementing the SMXQ -, the SMA - and the SMQMA policies in a net, the parameters m_i , b_i $\forall i = 1, 2, \dots, n$ could be seen as functions of the different loads and arrival rates on the transmission links. Thus the pure SMXQ -, SMA - and SMQMA policies require their parameters to be changed, when the different loads and/or the arrival rates do change.

Therefore let us define the policies above in a fixed form and call them "Fixed Sharing with Maximum Queues", FSMXQ, "Fixed Sharing with Minimum Allocation", FSMA, and "Fixed Sharing with Maximum Queues and Minimum Allocation", FSMQMA. When using these policies, we derive the parameters m_i and b_i for one specific load and arrival rate situation and assume that the parameters are not changed when load and/or arrival rate change.

Moreover, let us define "Fixed Complete Partitioning", FCP, in the same principle way.

As we will discuss in 5.4, Irland [40] has treated in particular the SMXQ policy. However, in his thesis he claims that $b_i = b$ $\forall i = 1, 2, \dots, n$. Thus, in order to make comparisons with his work, we define "Equal Sharing with Maximum Queues", ESMXQ and "Equal Sharing with Minimum Allocation", ESMA, and for those claim that $b_i = b$ $\forall i = 1, 2, \dots, n$ and $m_i = m$ $\forall i = 1, 2, \dots, n$ respectively.

Irland proposes another special case concerning the SMXQ- and the SMA policies. He calls them "the Square root rule" and therefrom "the Square root SMXQ" and "the Square root SMA". As we will see in the next chapter, the word Square root stems from the fact

that the parameters b_i and m_i respectively are derived when $\rho_i = 1 \forall i = 1, 2, \dots, n$ and thereby b_i and m_i are found as functions of $M/\sqrt{n}$.

5.4 Comparisons

In the span between no sharing at all, that is CP, and total sharing, CS, we have presented a number of restricted buffer sharing policies. It appears that the CS and the CP policies have serious drawbacks and though the total number of buffers should be shared, we must restrict the sharing. The limitations in the SMXQ policy do incorporate these ideas, i.e. it does not allow a specific type of packets to hog all the buffers. However, the SMXQ does not guarantee a full utilization of the servers under heavy traffic considerations [45],[47]. Among n input streams a minority could together hog all the buffers and thus preventing the other packet types to reach their transmission links. The SMA policy, however, does in principle, by means of its allocation of dedicated buffers, guarantee all packet types to reach their transmission links and thereby gives rise to a better total utilization of the outgoing transmission links. The SMQMA policy is of course a direct consequence of the discussion, that lead from the CS- to the SMXQ policy, now applied to the SMA policy on the m_0 buffers equally shared among all packets.

Under the SMXQ- and the SMA policies the global performance measures, as for instance the blocking probability, have in principle the same structure, but when we study those belonging to each of the outgoing transmission links, they appear in quite different shapes. The SMA policy for instance tends to contribute to better performance measures under heavily unbalanced load situations.

It is obvious that the CP- as well as the CS policy could be seen as a subset of the SMXQ-, the SMA- and the SMQMA policies. Take for instance the SMA policy and let $\sum_{i=1}^n m_i = M$ and we get the CP policy. Let $m_i = 0 \forall i = 1, 2, \dots, n$ and we have the CS policy.

Furthermore, if $n=2$, i.e. if we have two outgoing transmission links, then the SMA- and the SMXQ policies are identical. Let $b_1 = m_1 + m_0$ and $b_2 = m_2 + m_0$. If $n > 2$, the two policies are quite different. Latouche [61] roughly ranks four of our policies in order of increasing degree of sharing and if we include the SMQMA policy we get: CP, SMA, SMQMA, SMXQ and CS.

5.5 Previous analyses of restricted buffer sharing policies

During the last ten years much effort has been put into the analysis of computer communication networks with limited buffer space in the nodes.

Regarding the restricted buffer sharing policies SMA, SMXQ, SMQMA Kamoun [45] has made a pioneering effort in not only describing them, but furthermore he has cleared out the basic differences between performance measures derived through them. Besides his Ph D thesis he has together with Dr Leonard Kleinrock produced some essential articles in the field [47],[48]. Irland [40],[41],[42] has made excellent studies of computer communication networks and made pioneering studies of his Square root rule and the SMXQ policy. The works of Irland has been further developed by Latouche [61]. Foschini, Gopinath and Hayes [26] try to find "the optimal policy" and determine the form in the case when $n=2$ and $n=3$.

CHAPTER 6 - A PACKET SWITCHING NODE MODEL

6.1 Introduction

In this chapter we present a model of the packet switching node, where the occupancy of the output queue system is restricted according to one of the policies presented in chapter 5. After having in 6.2 repeated the fundamental packet switching principles, we state in 6.3 the basic assumptions behind our packet switching node model. In 6.4 we derive the steady state probability distribution, in 6.5 the blocking probabilities, and in 6.6 we discuss different cost functions used to optimize the parameters in our restricted buffer sharing policies. In 6.7 we present some sample results, in terms of blocking probabilities, buffer allocation, throughput and delays, derived under the CS- and SMA policies. Before we are ready to make conclusions in 6.9, we study in 6.8 the impact of offered load and arrival rate on the number of allocated buffers.

6.2 The packet switching node

In order to motivate our packet switching node model, let us repeat the principle function of the packet switching node.

In a packet switching computer communication network most computers are connected to the so called communication sub-network consisting of high speed transmission links and switching computers. The sub-network provides the delivery of packets from for instance host A to host B according to the Store-and-Forward principle (see figure 6.1). In host A a message is decomposed into packets, each of which has a maximum length, and these packets are delivered to

node 1. To each packet is a header attached with information of among other things the origin and the destination nodes. When an entire packet is received at node 1, it is placed in an output queue waiting for the, by the routing procedure chosen, transmission link to become free. After reaching for instance node 3, the Store-and-Forward principle is repeated until the packet reaches host B via node 5. When all the packets within the message have been delivered, possibly over different paths in the sub-network, the packets are re-assembled and so the user at host A can make use of a resource at host B.

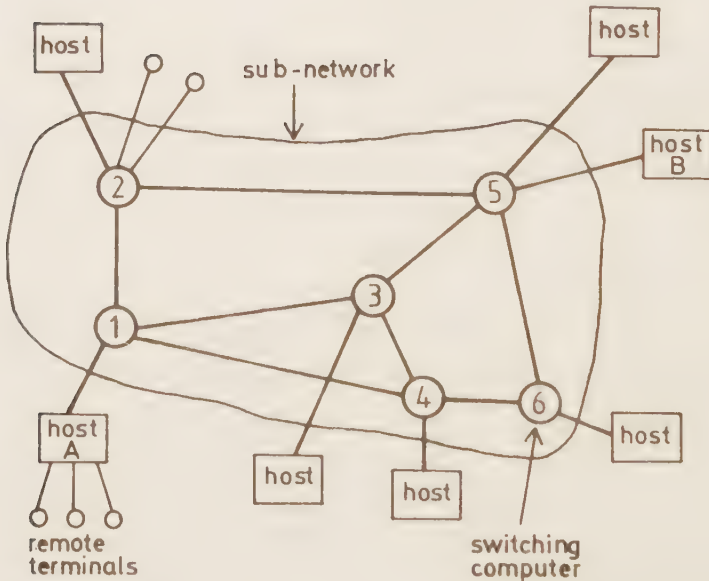


Figure 6.1

A packet's way through a node can briefly be described as follows: After arriving at the node, via one of the incoming transmission links, the packet is placed in an input queue, waiting for the routing procedure. The procedure chooses, according to the destination of the packet and in some cases to the traffic situation in the net, one outgoing transmission link and puts the packet in the corresponding output queue. We assume fix routing, which implies that the choice of the outgoing transmission link only depends on the destination of the packet.

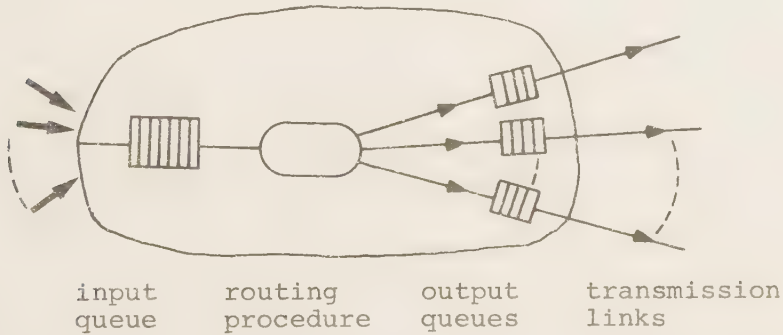


Figure 6.2

During transmission a copy of the packet is held in the output queue for retransmission if no acknowledgement signal, or a negative one, is received within a given time.

6.3 The problem in essence

As discussed in [40],[41], the execution of the routing procedure is much faster than the transmission and thus the routing procedure and the incoming queue are omitted from our model of the packet switching node.

In general, there are n outgoing transmission links leaving our node and we denote a packet, which will be sent over transmission link j a j -packet. The figure below shows the n different packet types, the outgoing queue or buffer system and the transmission links.

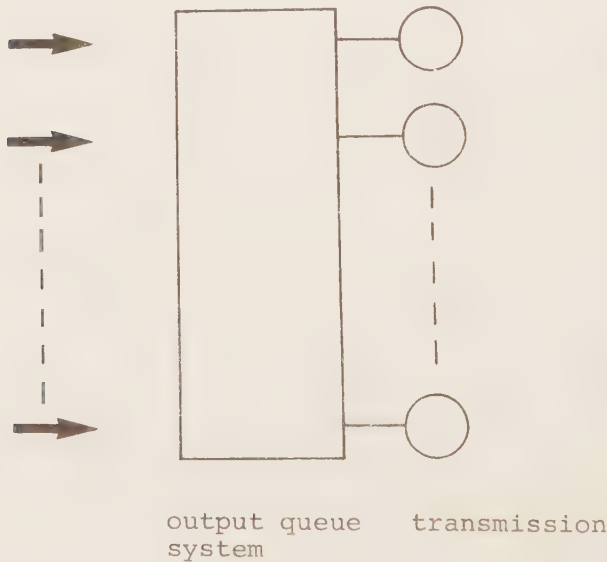


Figure 6.3

We assume furthermore, that the output queue system consists of M buffers, each of which can hold one packet irrespective of the length of the packet. The occupancy of the M buffers is restricted according to one of the restricted buffer sharing policies presented in chapter 5.

Kamoun [45], Gerla [30] and Fultz [29] discuss the questions that arise in conjunction with the fix routing assumptions. As our analytic tools do not, in our global packet switching node model, lend themselves to incorporate advanced adaptive routing policies (and not even non-advanced), we have to assume fix routing. Thus the validity of this assumption depends on the difference between fix and adaptive routing technique. However, the work of Fultz, though dealing with a special adaptive routing policy and network, shows a similarity in principle between the behaviour of a net using the fixed and the adaptive policy respectively.

As the adaptive routing policy requires information containing the net load situation, (i.e. the number of packets being sent between all adjacent nodes), which as a result implies an increase of total load, delay for instance tends to be higher.

Assume that j -packets arrive according to a Poisson stream with the intensity λ_j packets per second. We also assume that j -packet lengths are exponentially distributed with mean $1/\mu$ bits and that the capacity of output link j is C_j bits per second. This implies that the transmission times on the j -th link are exponentially distributed with mean $1/\mu C_j$ seconds.

Arriving packets, finding the output queue system "filled", according to the used restricted buffer sharing policy, are lost.

During transmission of a packet, a copy is held in the output queue system. Moreover, j -packets stored in the output queue system are transmitted in a FCFS basis.

Our assumption of exponentially distributed packet

lengths are not as invalid as it may seem at a first glance. Though messages, sent from one host computer to another, are divided into packets of fixed lengths (1024 bits in the ARPA network), measurements on the ARPA network have shown that the average size of a data message is some 250 bits [52]. As Kamoun [45] proposes, a better approximation would be to assume a truncated exponential message length distribution, but no closed form solution has been obtained [17].

In this chapter we restrict our derivations to the SMA- and CS policies as the SMA policy in itself contains both the CS and the CP policies and if $n=2$ is identic to the SMXQ policy.

6.4 The steady state probability distribution

Let $\underline{k} = (k_1, k_2, \dots, k_n)$ denote the state k_j j -packets in the output queue system ($\forall j = 1, 2, \dots, n$). Feasible states within the state space Ω are discussed in chapter 7. The states $\underline{k}$ in Ω form an n -dimensional ergodic Markov chain. To obtain the stable state solution $P(\underline{k})$ one can solve the global balance equations for the birth death process. This can be done with an n -dimensional z -transform and is done in appendix A for $n=2$ and the SMA policy, in order to show the technique in principle. The steady state solution

$$P(k_1, k_2, \dots, k_n) = P(\underline{0}) \prod_{j=1}^n \rho_j^{k_j} \quad (6.1)$$

where $P(\underline{0}) = P(0, 0, \dots, 0)$ and $\rho_j = \frac{\lambda_j}{\mu C_j}$

can also be obtained from the local balance equations. The solution obtained from these must of course satisfy the global balance equations [47].

The normalization constant $P(\underline{0})$ is given by the conservation relation

$$\sum_{\Omega} P(\underline{k}) = 1 \quad (6.2)$$

$$P(\underline{0})^{-1} = \sum_{\Omega} \prod_{j=1}^n \rho_j^{k_j} . \quad (6.3)$$

6.5 Blocking probabilities

Under the SMA policy let k_j^+ denote the number of j -packets in the m_0 common buffers. Thus

$$k_j^+ = \max(0, k_j - m_j) \quad (6.4)$$

or

$$k_j^+ = (k_j - m_j)^+ \quad (6.5)$$

An arriving j -packet is lost if

$$k_j \geq m_j \quad (6.6)$$

and

$$\sum_{j=1}^n k_j^+ = m_0$$

The probability of losing an arriving j -packet, B_j , can be written

$$B_j = P(k_j \geq m_j, \sum_{j=1}^n k_j^+ = m_0) \quad (6.7)$$

B_j can be seen as the sum over all $k = 0, 1, \dots, m_0$, of the probability that there are $m_j + k$ j -packets in the total output queue system and $m_0 - k$ "non j -packets" in the m_0 common buffers.

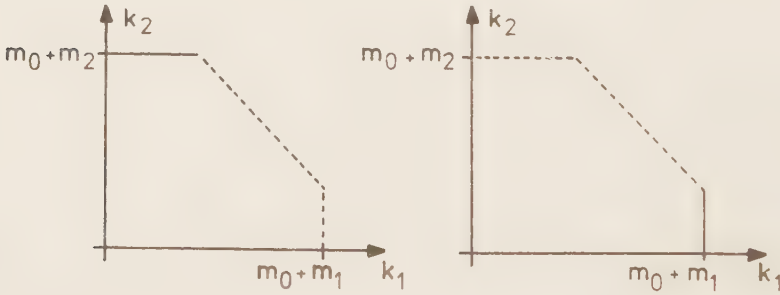
$$B_j = \sum_{k=0}^{m_0} \sum_{\substack{i \\ i \neq j}} \sum_{k_i^+ = m_0 - k} P(k_1, k_2, \dots, k_{j-1}, m_j + k, k_{j+1}, \dots, k_n); \quad (6.8)$$

where

$$m_0 = M - \sum_{j=1}^n m_j \quad (6.9)$$

In chapter 7 we propose a recursive technique to compute the normalization constant and the blocking probabilities in the general n -dimensional case.

In the case when we have two outgoing transmission links ($n=2$), B_1 and B_2 are, respectively, given by the sum over the states belonging to the dotted lines in the figures below.



$B_1 = \sum_{\phi_1} P(\underline{k})$; ϕ_1 : the broken line

$B_2 = \sum_{\phi_2} P(\underline{k})$; ϕ_2 : the broken line

Figure 6.4

When ρ_1, ρ_2 and the number 1 are all distinct, B_1 and B_2 are given by the following expressions

$$B_1 = \left(\rho_1^{m_1+m_0} \frac{(1-\rho_2^{m_2+1})}{(1-\rho_2)} + \rho_1^{m_1} \rho_2^{m_2+1} \frac{(\rho_1^{m_0} - \rho_2^{m_0})}{(\rho_1 - \rho_2)} \right) \cdot P(0,0) \quad (6.10)$$

$$B_2 = \left(\rho_2^{m_2+m_0} \frac{(1-\rho_1^{m_1+1})}{(1-\rho_1)} + \rho_2^{m_2} \rho_1^{m_1+1} \frac{(\rho_2^{m_0} - \rho_1^{m_0})}{(\rho_2 - \rho_1)} \right) \cdot P(0,0) \quad (6.11)$$

where

$$P(0,0)^{-1} = \frac{(1-\rho_1^{m_1+m_0+1})(1-\rho_2^{m_2+1})}{(1-\rho_1)(1-\rho_2)} + \frac{(1-\rho_2^{m_0})\rho_2^{m_2+1}}{(1-\rho_1)(1-\rho_2)} - \rho_1^{m_1+1} \rho_2^{m_2+1} \frac{(\rho_1^{m_0} - \rho_2^{m_0})}{(1-\rho_1)(\rho_1 - \rho_2)} \quad (6.12)$$

In the case when we have three outgoing transmission links ($n=3$), and thus the SMA and the SMXQ policies are quite different, B_1 is given as the sum over the states belonging to the shadowed surface in figure 6.5.

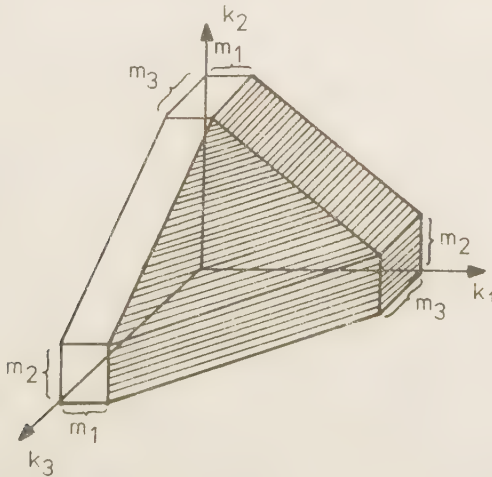


Figure 6.5

6.6 Cost functions

6.6.1 Cost functions - of what?

In order to obtain an optimal allocation of the number of dedicated buffers, we form a cost function, which should be a function of one of the performance measures we are interested in optimizing.

Thus, when using the SMA policy, the result of the optimization could yield the CP or the CS policies if the result of the optimization turned out to be $m_0 = 0$ or $m_0 = M$ respectively. However, in our earlier discussions and as we will see later, the CP policy causes high blocking probabilities and thus, as it blocks out too many packets, mean delay for the non blocked packets is shorter than the delays obtained as a result of other policies.

We can even form a hypothetical policy that blocks out most of the arriving packets and which thus gives the few non blocked packets a delay that consists of almost only the corresponding transmission times.

This leads to the conclusion that a cost function based on delays only is not useful.

We have primarily used as cost function "the number of lost j -packets per second", $\forall j = 1, 2, \dots, n$. A minimization of the cost function $f_1(\underline{m})$, with respect to the number of dedicated buffers, gives us the optimal allocation. The cost function $f_1(\underline{m})$ gives us

$$f_1(\underline{m}) = f_1(m_1, m_2, \dots, m_n) = \sum_{j=1}^n \lambda_j B_j \quad (6.13)$$

a minimum value of the number of lost packets per second, but, however, the given allocation may favour for instance

j_1 -packets before j_2 -packets. Another cost function, $f_2(\underline{m})$, though not minimizing the total number of lost packets per second, leads to that the number of lost packets per second will be better spread among the different types of packets.

$$f_2(\underline{m}) = \sum_{j=1}^n (\lambda_j B_j)^2 \quad (6.14)$$

In other cases a plausible cost function could contain weight factors in order to favour one or several packet types before others. The cost function $f_3(\underline{m})$ is used in chapter 9 in order to increase total throughput in a net.

$$f_3(\underline{m}) = \sum_{j=1}^n p_j \lambda_j B_j \quad (6.15)$$

Having in mind the form of B_1 and B_2 in the previous section, one can easily understand that the minimization of the cost function, with respect to the vector $\underline{m}$, has to be done numerically. Looking at the elements in the vector $\underline{m}$ as real variables, it is possible, after some simplifications and in some special cases, to obtain the optimal allocation through differentiation.

$$\frac{\partial f(\underline{m})}{\partial m_j} = 0 \quad \forall j = 1, 2, \dots, n. \quad (6.16)$$

6.6.2 The square-root rule

As mentioned in chapter 5, we may optimize our cost function when $\rho_i = 1 \quad \forall i = 1, 2, \dots, n$ and use the obtained parameters, say the number of dedicated buffers under the SMA policy, in other load situations. This technique was proposed independently by Irland [40] and Körner [57]. Irland uses the technique in combination with the SMXQ policy.

If we use this technique in combination with the SMA policy, the expressions for the blocking probabilities and the normalization constant collapse into (for the two dimensional case)

$$B_1 = (M - m_1 + 1)P(0,0) \quad (6.17)$$

$$B_2 = (M - m_2 + 1)P(0,0) \quad (6.18)$$

$$P(0,0)^{-1} = m_1 m_2 + M(1 + m_0) + 1 + \frac{m_0}{2}(1 - m_0); \quad (6.19)$$

Let

$$f(\underline{m}) = \lambda_1 B_1 + \lambda_2 B_2 \quad (6.20)$$

and

$$B_1(m_1, m_2) = A_1(m_1, m_2) \cdot C(m_1, m_2) \quad (6.21)$$

$$B_2(m_1, m_2) = A_2(m_1, m_2) \cdot C(m_1, m_2) \quad (6.22)$$

where

$$C(m_1, m_2) = P(0,0) \quad (6.23)$$

Thus

$$f(\underline{m}) = \lambda_1 A_1(m_1, m_2) C(m_1, m_2) + \lambda_2 A_2(m_1, m_2) C(m_1, m_2) \quad (6.24)$$

$$\begin{aligned} \frac{\partial f(\underline{m})}{\partial m_1} = & \lambda_1 \left(\frac{\partial A_1(m_1, m_2)}{\partial m_1} \cdot C(m_1, m_2) + \frac{\partial C(m_1, m_2)}{\partial m_1} \cdot A_1(m_1, m_2) \right) + \\ & + \lambda_2 \left(\frac{\partial A_2(m_1, m_2)}{\partial m_1} \cdot C(m_1, m_2) + \frac{\partial C(m_1, m_2)}{\partial m_1} \cdot A_2(m_1, m_2) \right) \end{aligned} \quad (6.25)$$

$$\begin{aligned} \frac{\partial f(\underline{m})}{\partial m_2} = & \lambda_1 \left(\frac{\partial A_1(m_1, m_2)}{\partial m_2} \cdot C(m_1, m_2) + \frac{\partial C(m_1, m_2)}{\partial m_2} \cdot A_1(m_1, m_2) \right) + \\ & + \lambda_2 \left(\frac{\partial A_2(m_1, m_2)}{\partial m_2} \cdot C(m_1, m_2) + \frac{\partial C(m_1, m_2)}{\partial m_2} \cdot A_2(m_1, m_2) \right); \end{aligned} \quad (6.26)$$

Let us write

$$\frac{\partial C(m_1, m_2)}{\partial m_i} = -C^2 \frac{\partial \left(\frac{1}{C(m_1, m_2)} \right)}{\partial m_i} \quad i = 1, 2 \quad (6.27)$$

Let furthermore $C_1 = C_2 \Rightarrow \lambda_1 = \lambda_2 = \lambda$. As m_1 and m_2 now are real, the two blocking probabilities will be equal and thereby $m_1 = m_2 = m$.

Let us write

$$B_1 = B_2 = A(m) \cdot C(m) \quad (6.28)$$

The equations (6.25) and (6.26) then collapse into

$$\frac{\partial f(m)}{\partial m} = 2\lambda \left(\frac{\partial A(m)}{\partial m} \cdot C(m) - A(m) \cdot C(m)^2 \frac{\partial \left(\frac{1}{C(m)} \right)}{\partial m} \right); \quad (6.29)$$

and we find the optimal m from

$$\frac{\partial A(m)}{\partial m} \cdot C(m) - A(m) \cdot C(m)^2 \frac{\partial \left(\frac{1}{C(m)} \right)}{\partial m} = 0 \quad (6.30)$$

$$\Rightarrow \frac{\partial A(m)}{\partial m} \cdot \frac{1}{C(m)} - A(m) \frac{\partial \left(\frac{1}{C(m)} \right)}{\partial m} = 0 \quad (6.31)$$

We know that $B = (B_1 = B_2) = A(m) \cdot C(m)$ where

$$A(m) = M - m + 1 \quad (6.32)$$

$$C(m)^{-1} = m^2 + M(1 + M - 2m) + 1 + \frac{M - 2m}{2}(1 - M + 2m) \quad (6.33)$$

$\Rightarrow$

$$C(m)^{-1} = \frac{M^2}{2} + \frac{3M}{2} - m^2 - m + 1; \quad (6.34)$$

$$\frac{\partial A(m)}{\partial m} = -1 \quad (6.35)$$

$$\frac{\partial \left(\frac{1}{C(m)} \right)}{\partial m} = -2m - 1 \quad (6.36)$$

If we put these last two equations into (6.31), we get

$$-1\left(\frac{M^2}{2} + \frac{3M}{2} + 1 - m - m^2\right) + (1 + 2m)(1 + M - m) = 0 \quad (6.37)$$

$\Rightarrow$

$$m^2 + m(2 + 2M) + \frac{M}{2}(1 + M) = 0 \quad (6.38)$$

$\Rightarrow$

$$m = (M + 1) \left(\pm\right) \frac{1}{2} \sqrt{2M^2 + 6M + 4} \quad (6.39)$$

The result can approximately be seen as

$$m \approx M\left(1 - \frac{1}{\sqrt{2}}\right) \quad (6.40)$$

If we compare this result with the square root rule, proposed by Irland, he finds $b_i = b$, $i = 1, 2$

$$b \approx \frac{M}{\sqrt{2}} \quad (6.41)$$

we find the two policies being identic as

$$b = M - m; \quad (6.42)$$

6.7 Sample results

In this chapter we restrict the presentation of our results to one unbalanced load situation, where one of the packet streams is varied and the others are held constant and to one fairly balanced load situation, where all packet streams are simultaneously increased and have almost equal intensity.

In 6.7.1 we present the individual blocking probabilities B_i ($i = 1, 2, \dots, n$) as well as the total blocking probabilities $B = (1/\lambda) \sum \lambda_i B_i$ where $\lambda = \sum \lambda_i$. In section 6.7.2 the allocation scheme, i.e. the number of dedicated buffers, is presented as a function of load under the SMA policy. In order to compare the SMA and the CS policies we present a cost reduction function in 6.7.3. In 6.7.4 and 6.7.5 throughput and transit delay are presented. The FSMA and the CS policies are compared in 6.7.6 in terms of the number of lost packets per second and finally in 6.7.7 we compare results derived under the ESMXQ (or ESMA as $n = 2$) and the SMXQ (or SMA) policies.

When studying the results presented in this chapter, one has to be extra observant and critical. It is easy to "forget" one of the parameters when studying a diagram. Furthermore the benefits one policy gives in comparison with another do often increase or decrease with load. Comparing for instance the SMA and the CS policies, the number of lost packets are almost equal for the two policies under a light load, but do diverge as load increases.

Why do we then deal with restricted buffer sharing policies, when we know that memory space today is not expensive and will become even cheaper in the future? The benefits of restricted buffer sharing policies are

not, as we have discussed earlier, limited to a decrease of the number of lost packets and transit delays, but they also yield a better network performance in terms of protecting the entire communication net from being locked up in abnormal situations. These situations could for instance arise as a result of a software failure in one of the nodes causing a dramatic increase of the number of packets leaving that node.

6.7.1 Blocking probabilities

In figure 6.6 we find the blocking probabilities

$$B = \frac{\lambda_1 B_1 + \lambda_2 B_2}{\lambda_1 + \lambda_2} \quad (6.43)$$

drawn as a function of load offered link 2. We find the principle behaviour, earlier discussed in chapter 5, of the three buffer sharing policies CP, CS and SMA. The curves correspond to a fairly balanced load where $\rho_1 = \rho_2 = 0.1$. As seen in the figure, the blocking probabilities corresponding to the CP policy are rather high when load is small but very acceptable for heavy loads. On the contrary blocking probabilities corresponding to the CS policy are acceptable for a low offered traffic, but too high when load increases. As we will see in the next section, the "optimal policy", SMA, will allocate buffers according to the CS policy and the CP policies for low and high loads respectively.

In figure 6.7 the different link blocking probabilities are shown for the same load situation as in figure 6.6. As $B_1 = B_2 = B$ in the CS case they are omitted. The steps in the non linear curves are of course due to changes in the number of allocated buffers. We find the SMA policy favouring packets bound for transmission link 1 as

load on this link is smaller than that of link 2. This effect will be clearly seen under an unbalanced load situation and is further discussed in section 6.8. The reason for the curves corresponding to the CP policy being linear is that the CP policy allocates 5 buffers to each output link in the whole interval.

In figures 6.8 to 6.11 the blocking probabilities are drawn as a function of ρ_2 under an unbalanced load situation. In figures 6.8 and 6.9 ρ_1 is held constant and equal to 0.5 and in 6.10 and 6.11 ρ_1 is equal to 0.9. Comparing figure 6.8 with 6.6 we find the CS policy to behave worse compared with the SMA policy in the unbalanced load situation. In figures 6.9 and 6.11 we find that the SMA policy favours packets bound for the least loaded transmission link. Moreover we find the magnitudes of the individual blocking probabilities to be equal, as they should, when $\rho_1 = \rho_2$ (see fig 6.11). This is of course intuitively pleasing, but one could state a hypothetical case where, though the two loads are equal, one of them, no matter which, should be allocated more buffers, than the other.

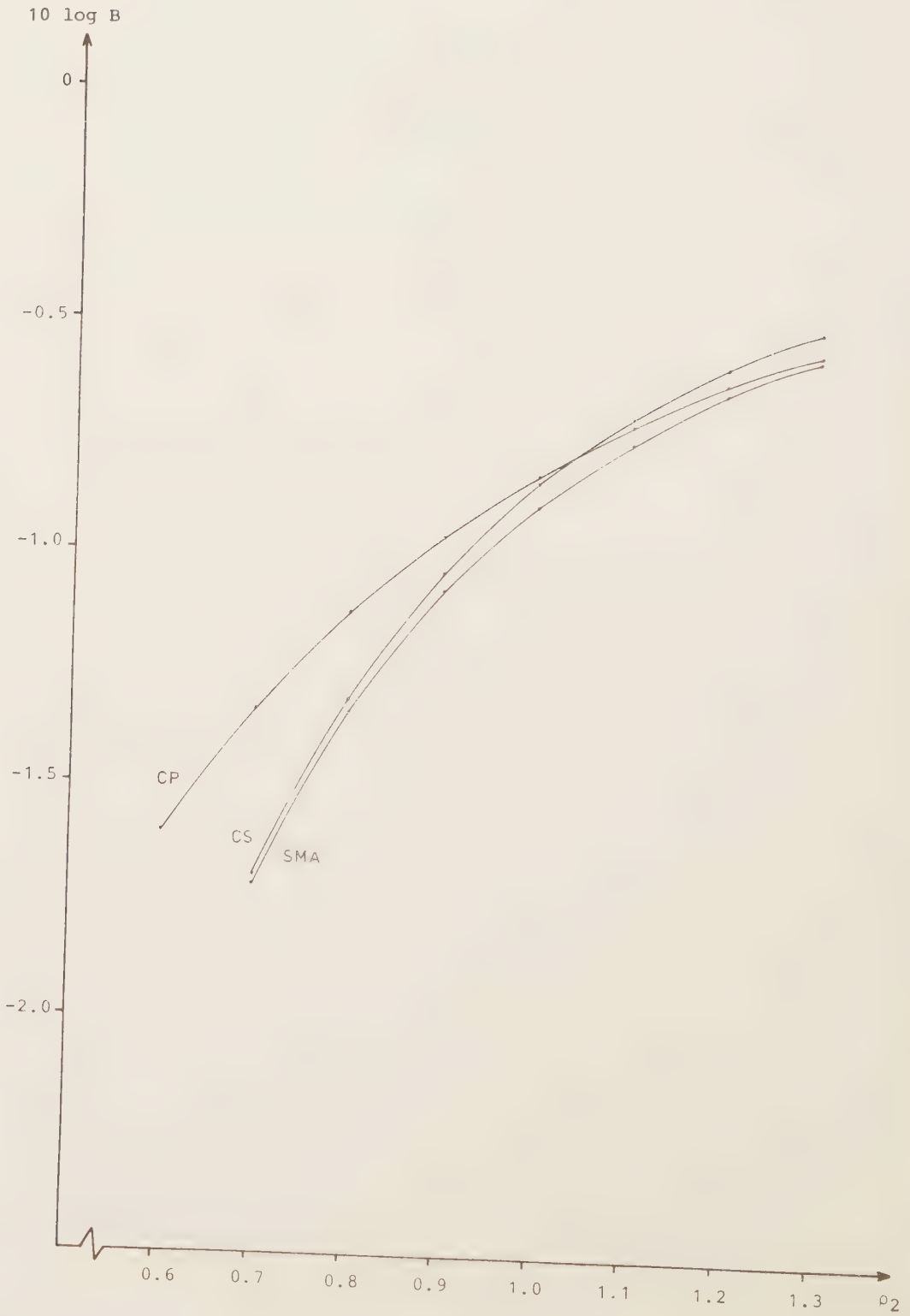


Figure 6.6 Fairly balanced load, $\rho_1 = \rho_2 - 0.1$, $C_1 = C_2$

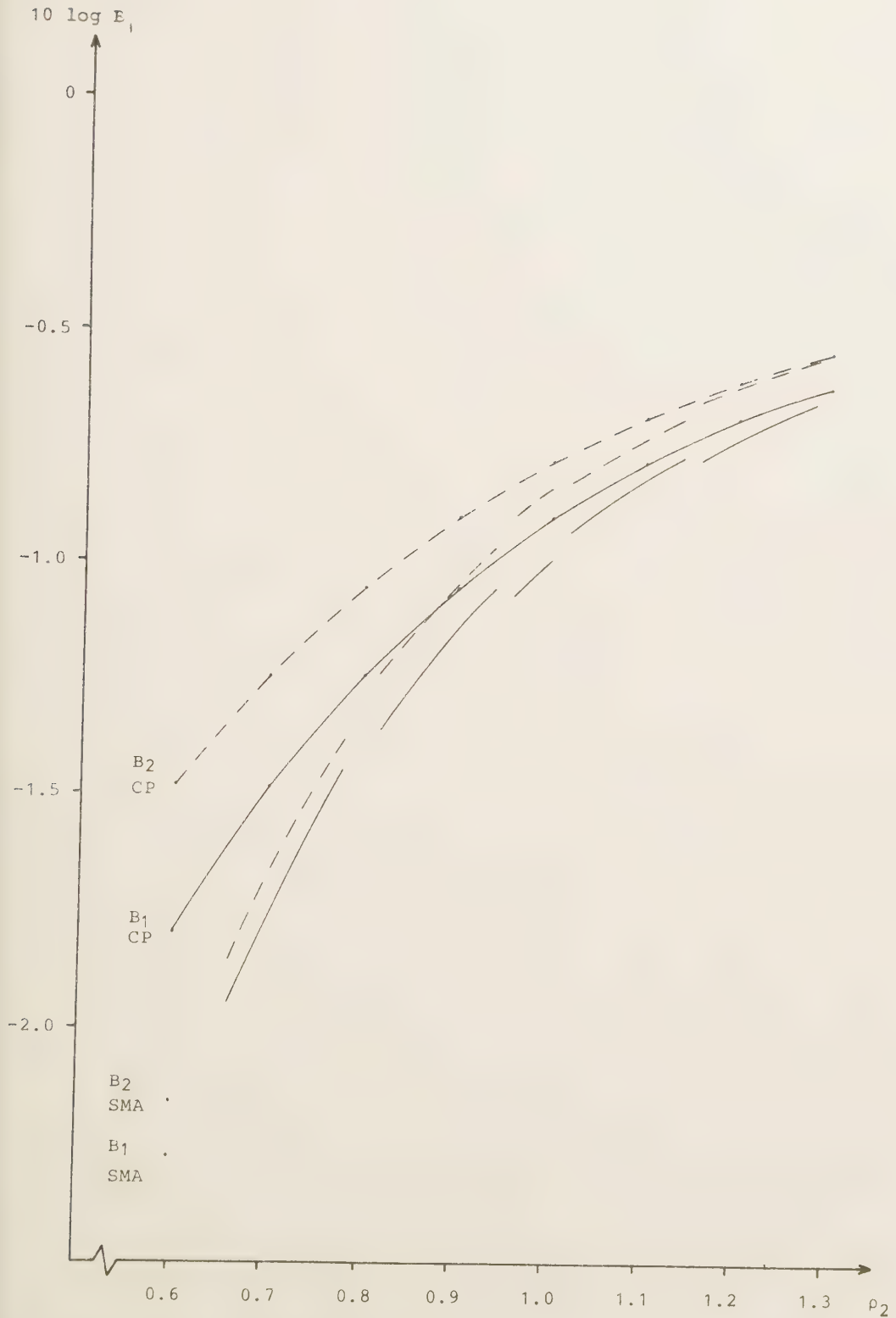


Figure 6.7 Fairly balanced load, $\rho_1 = \rho_2 - 0.1$, $C_1 = C_2$

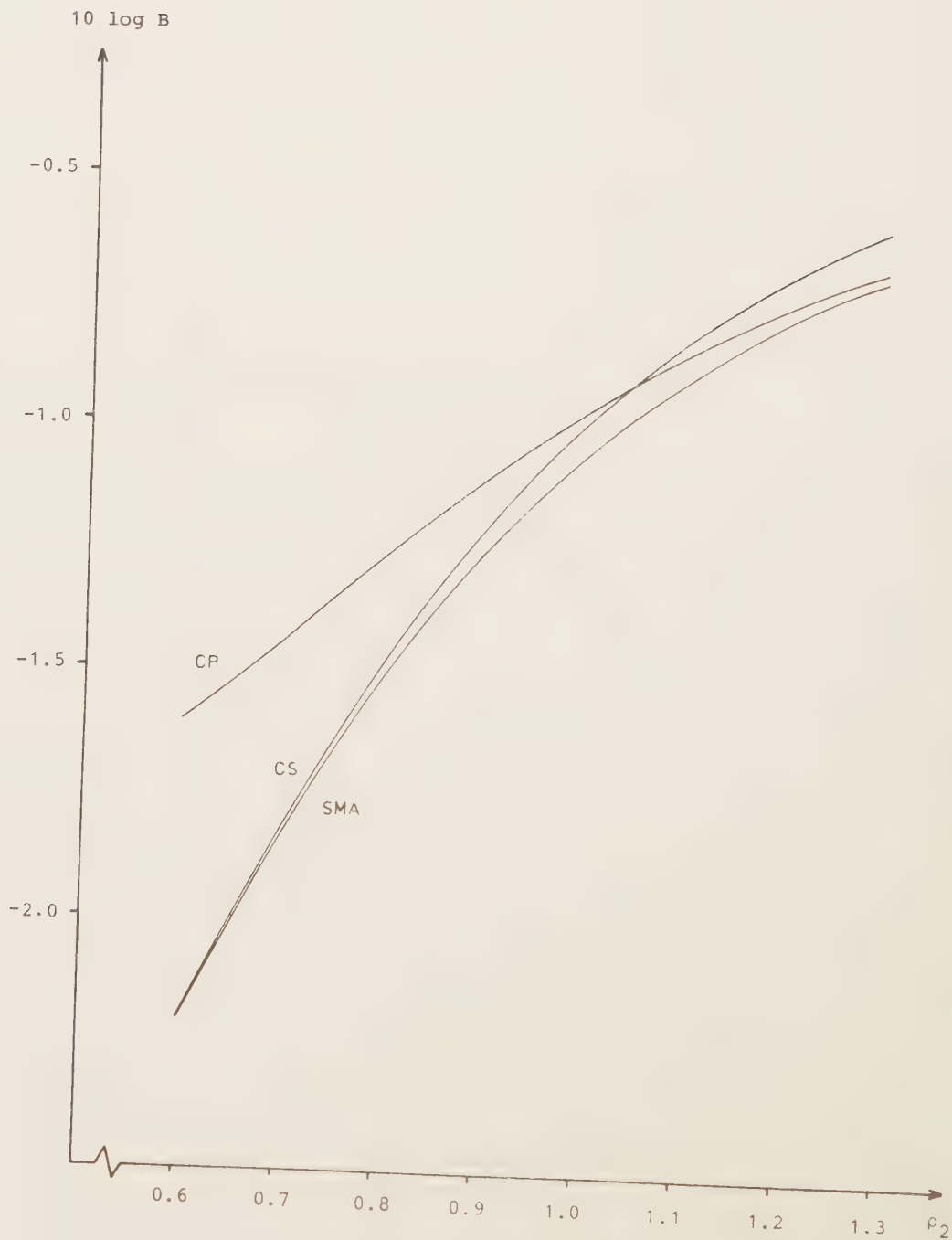


Figure 6.8 Unbalanced load, $\rho_1 = 0.5$, $C_1 = C_2$

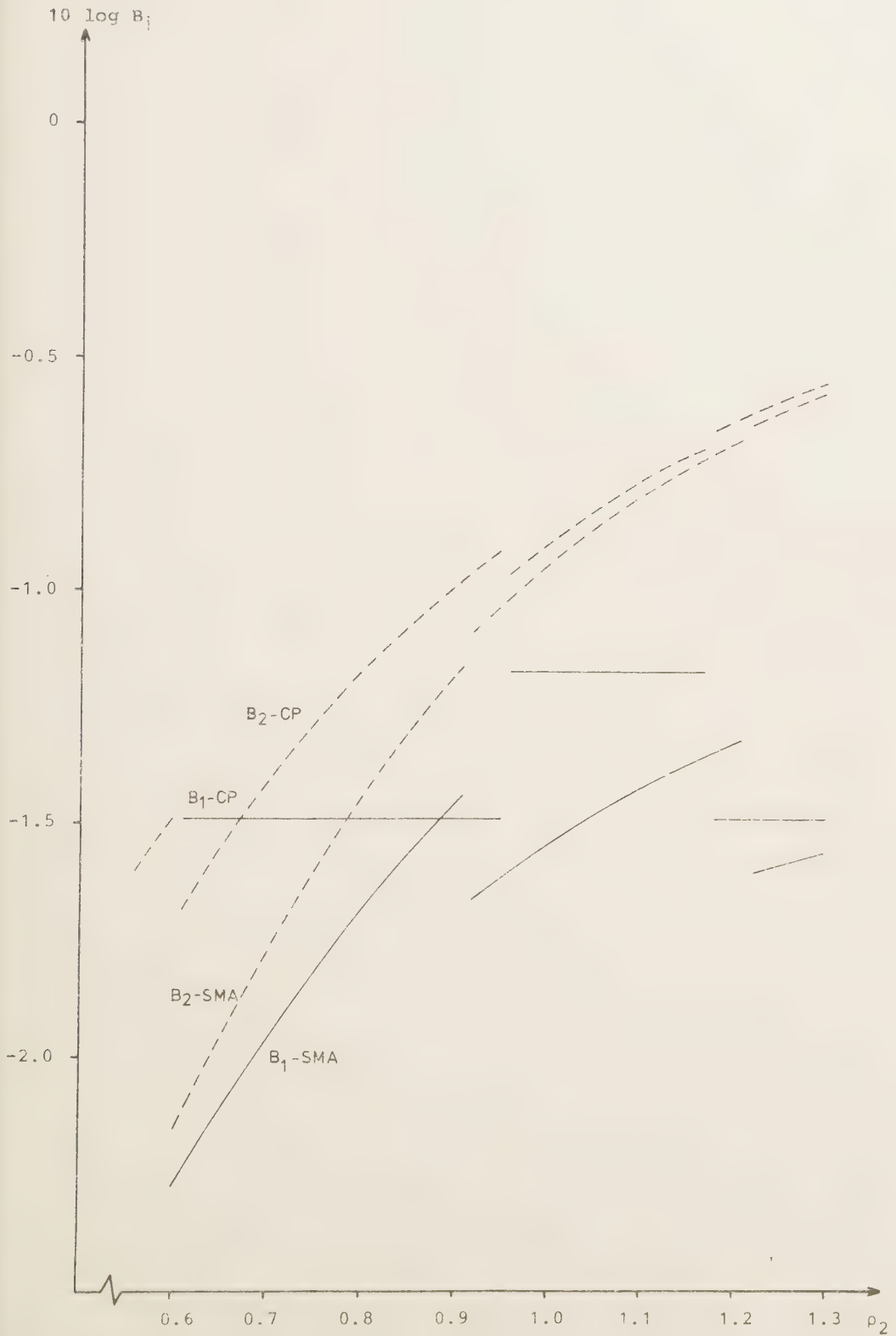


Figure 6.9 Unbalanced load, $\rho_1 = 0.5$, $C_1 = C_2$

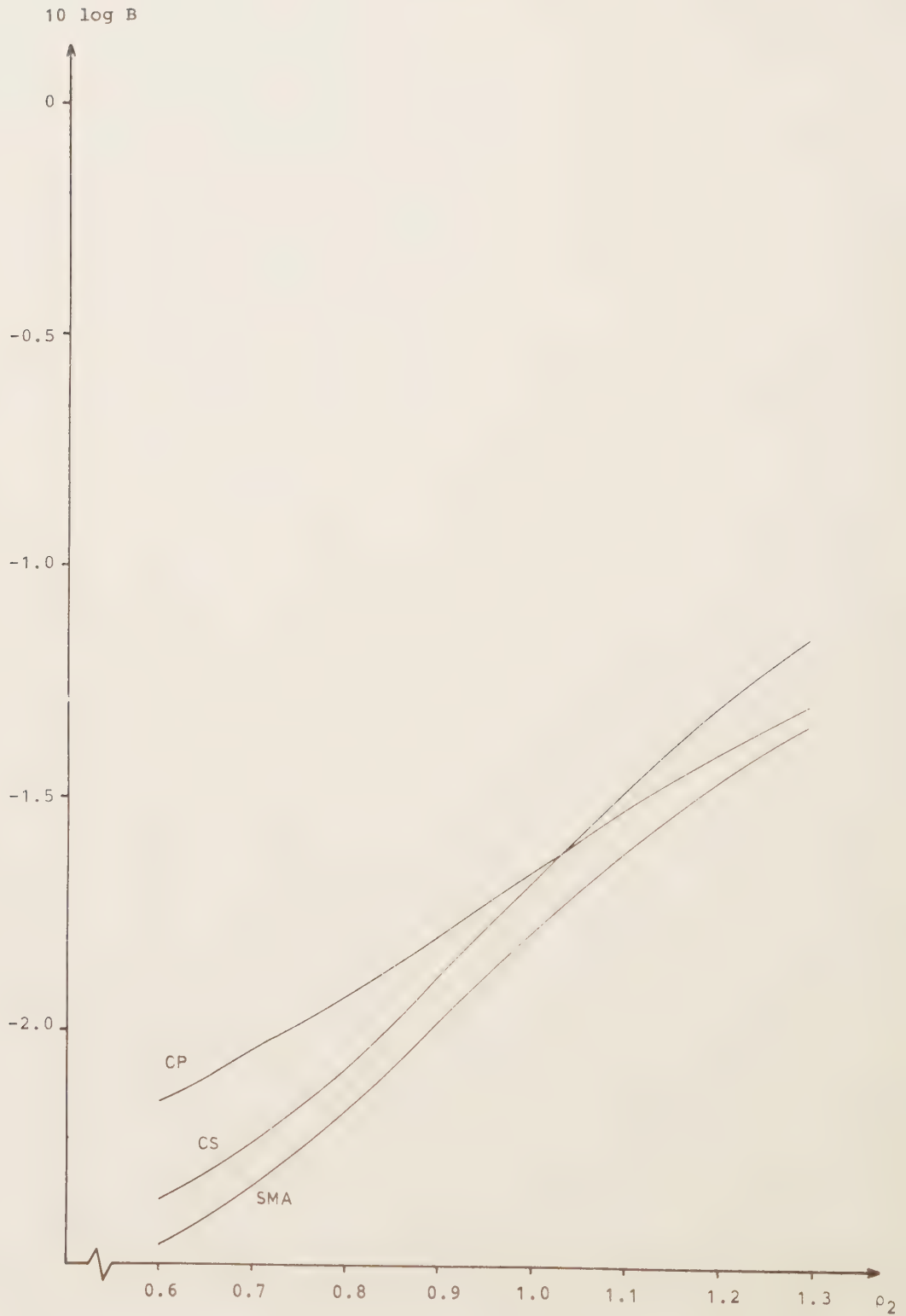


Figure 6.10 Unbalanced load, $\rho_1 = 0.9$, $C_1 = C_2$

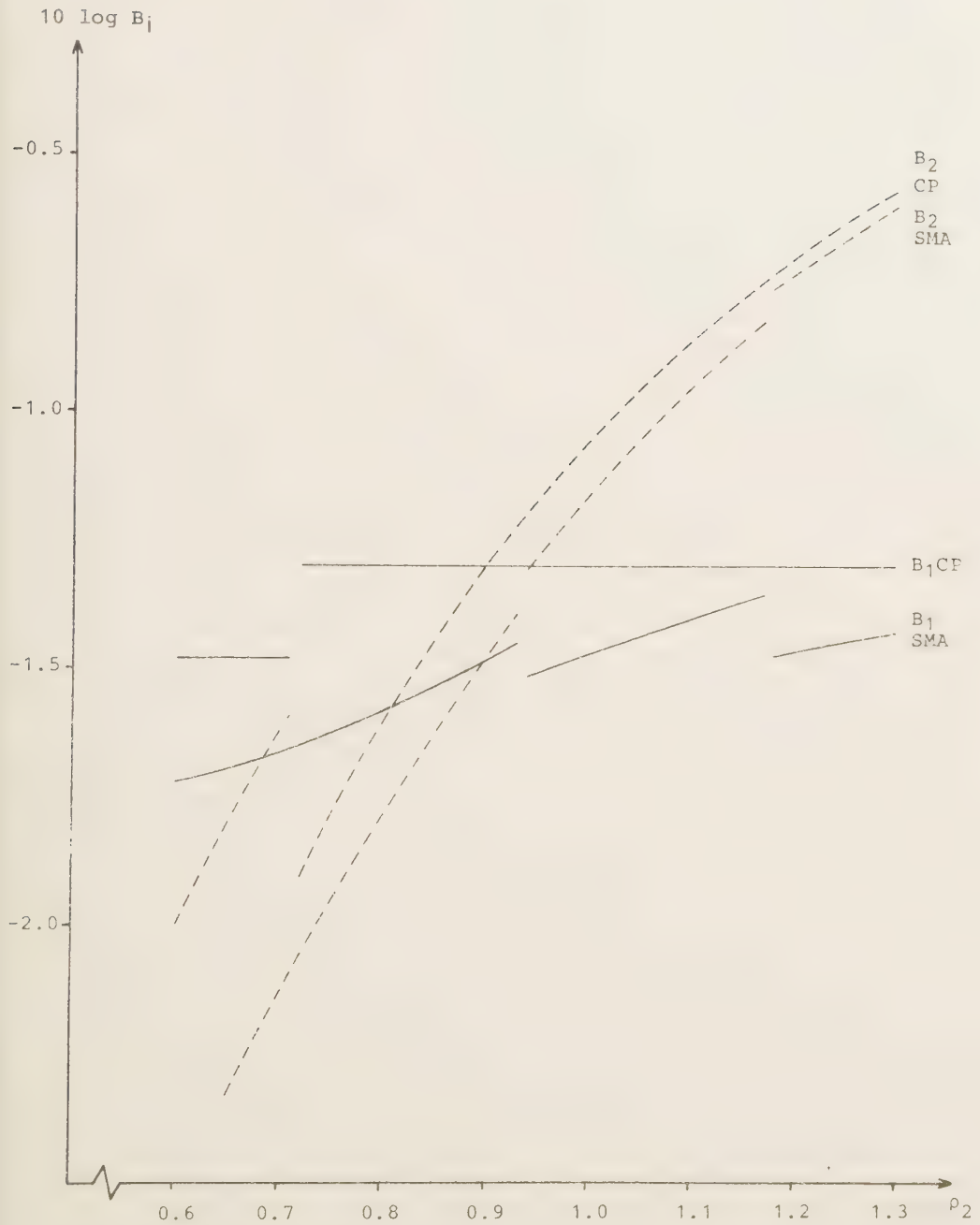


Figure 6.11 Unbalanced load, $\rho_1 = 0.9$, $C_1 = C_2$

Figures 6.12 to 6.15 show the blocking probabilities corresponding to a node with three outgoing transmission links. Figures 6.12 and 6.13 show the total and the individual blocking probabilities under our unbalanced load situation and 6.14 and 6.15 the appropriate measures under our "fairly" balanced load situation. Again we find the CP policy behaving poorly under light load and the same goes for the CS policy under heavy load. In figure 6.13 we find the SMA policy protecting 1- and 2-packets, when the stream of 3-packets increases. The leaps in the curves correspond to a change in the allocation of the number of dedicated buffers. In section 6.5.2 the corresponding allocation scheme is given. Furthermore, when comparing the CS and the SMA policies we find the former doing worse in the unbalanced load situation than in the balanced one.

In chapter 10, where we apply our restricted buffer sharing policies on a complete net, we will derive results not only under the CS, the CP and the SMA policies, but for the FSMA, the ESMA and the FCP policies as well.

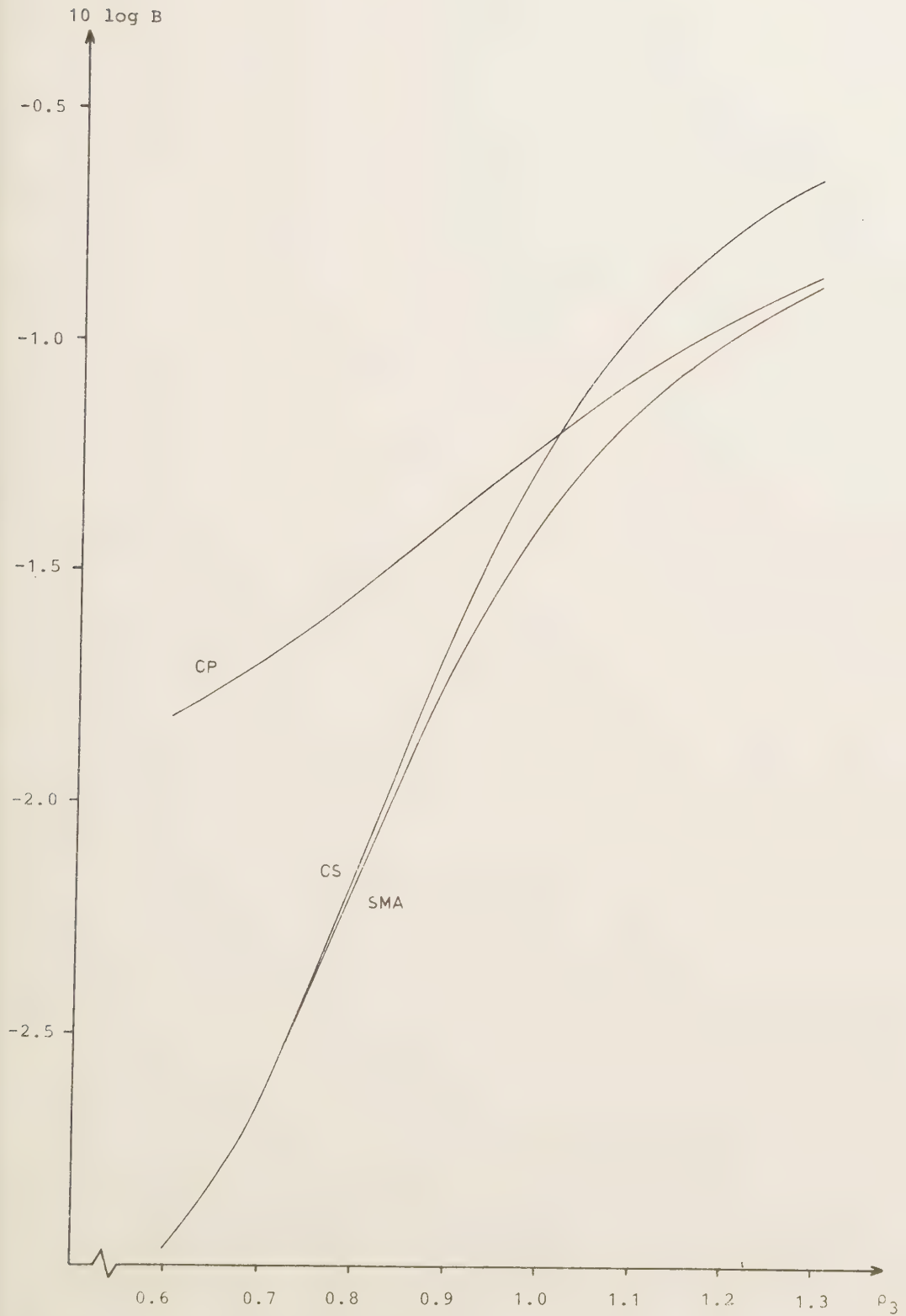


Figure 6.12 Unbalanced load, $\rho_1 = 0.5$, $\rho_2 = 0.7$, $C_1 = C_2 = C_3$

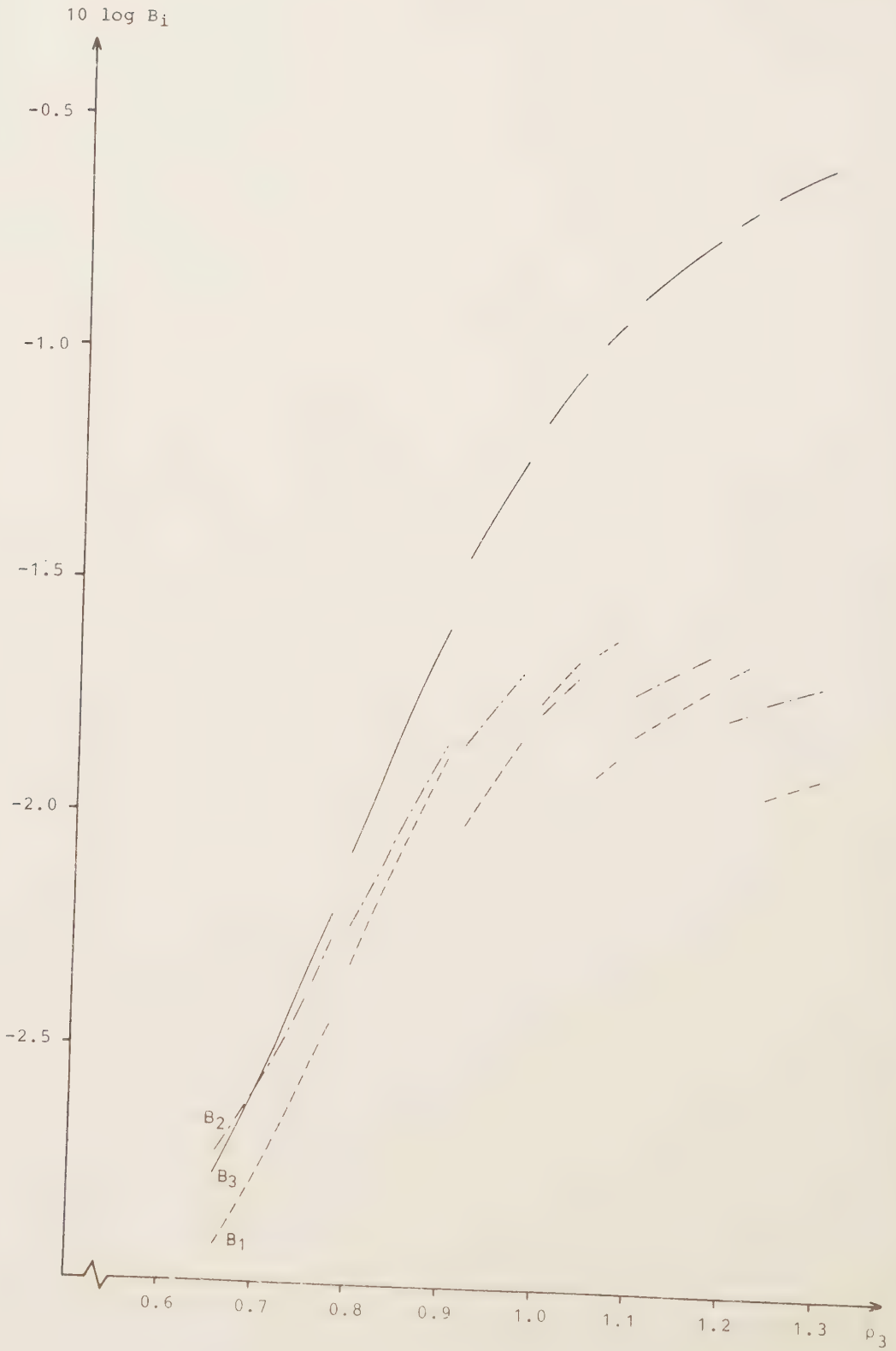


Figure 6.13 SMA, unbalanced load, $\rho_1 = 0.5$, $\rho_2 = 0.7$, $C_1 = C_2 = C_3$

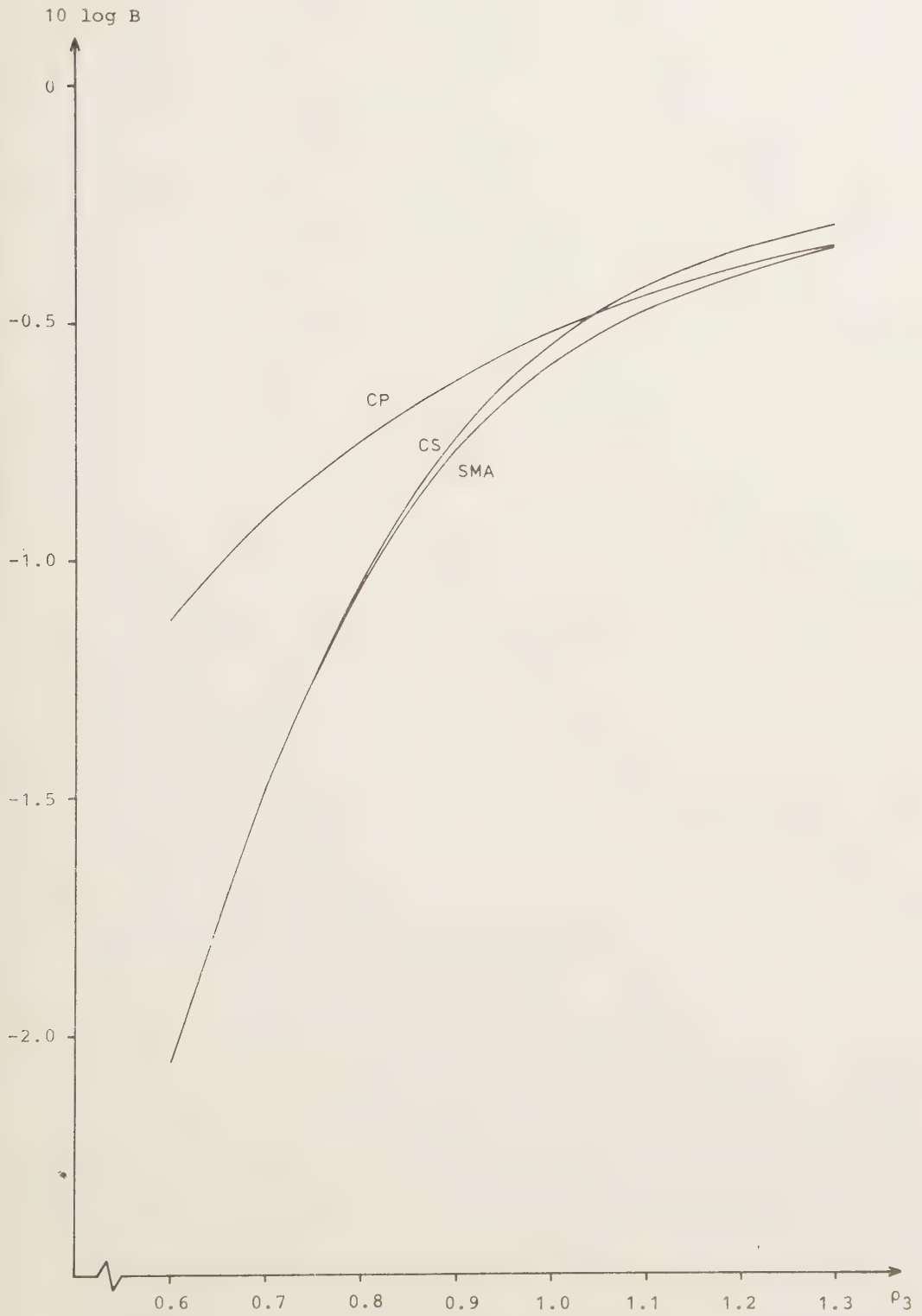


Figure 6.14 Fairly balanced load, $\rho_2 = \rho_3 - 0.1$, $\rho_1 = \rho_3 - 0.2$,
 $C_1 = C_2 = C_3$

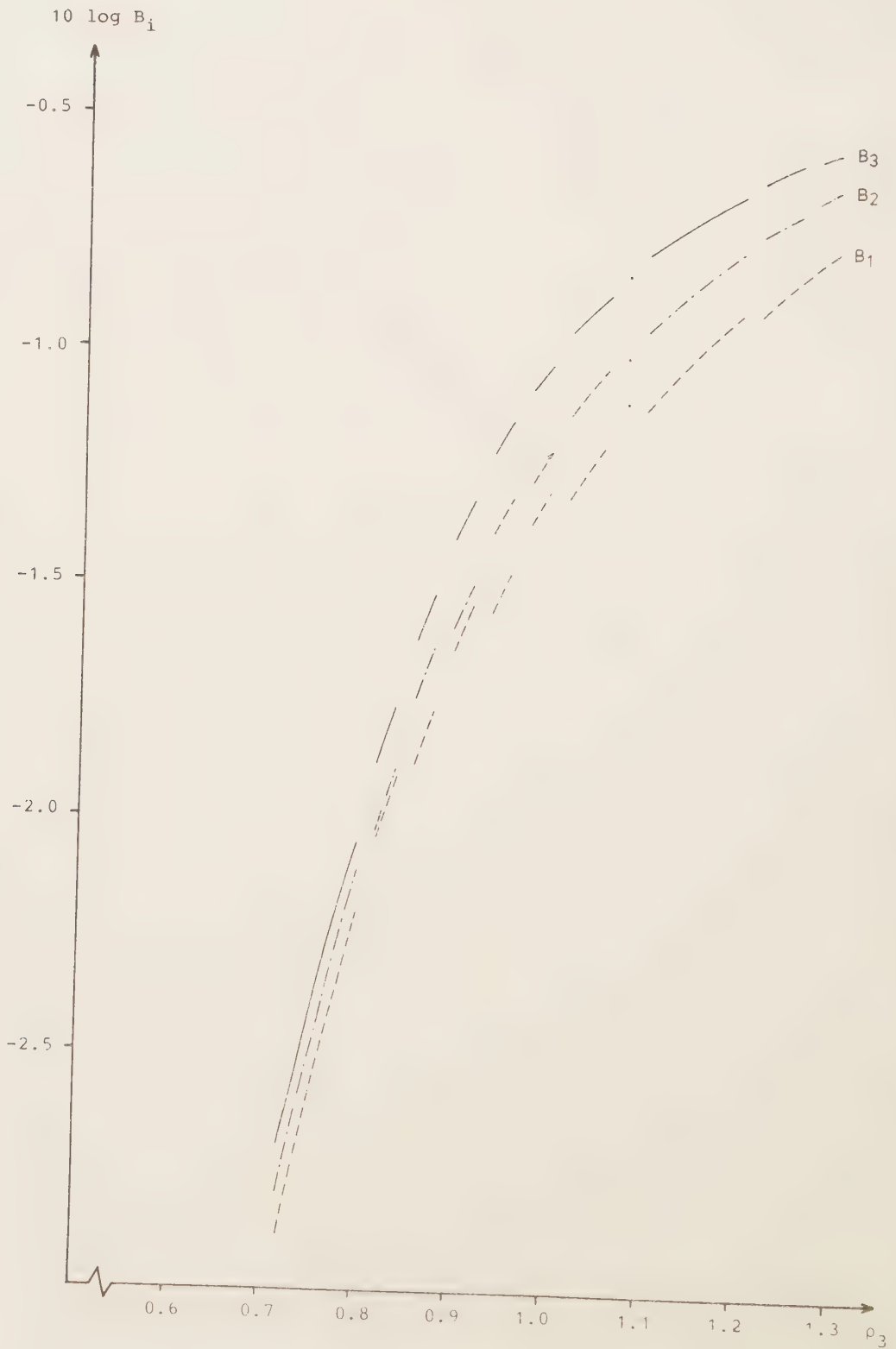


Figure 6.15 SMA, fairly balanced load, $\rho_2 = \rho_3 - 0.1$,

$$\rho_1 = \rho_3 - 0.2, C_1 = C_2 = C_3$$

6.7.2 Buffer allocation

In this section we present results concerning the number of dedicated buffers under the SMA policy. The presented results correspond to the load situations used in the previous section (6.7.1).

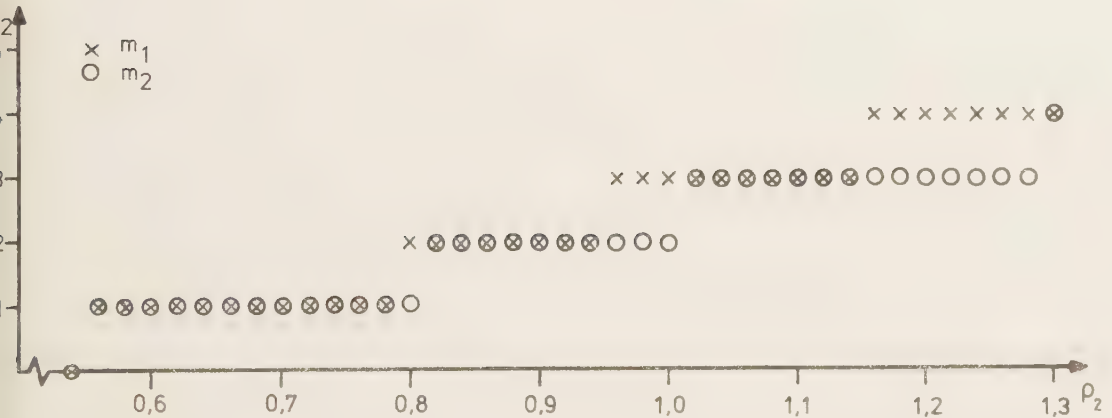


Figure 6.16 Fairly balanced load, $\rho_1 = \rho_2 - 0.1$

In figure 6.16 we find that the SMA policy allocates more dedicated buffers to 1-packets than to 2-packets when $\rho_1 < \rho_2$. When load increases packets bound for the most loaded transmission link tend to hog most of the buffers and thus prevent packets bound for the other transmission links to reach their link. This basic effect of

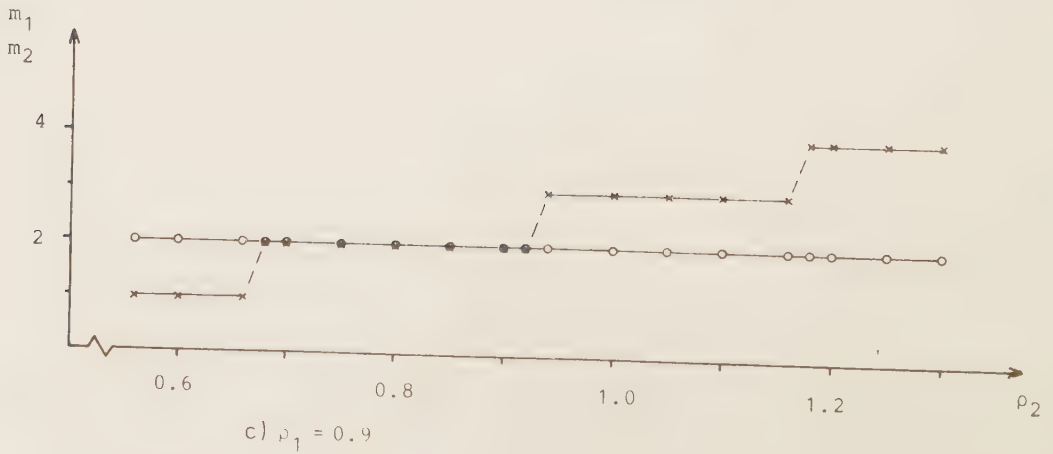
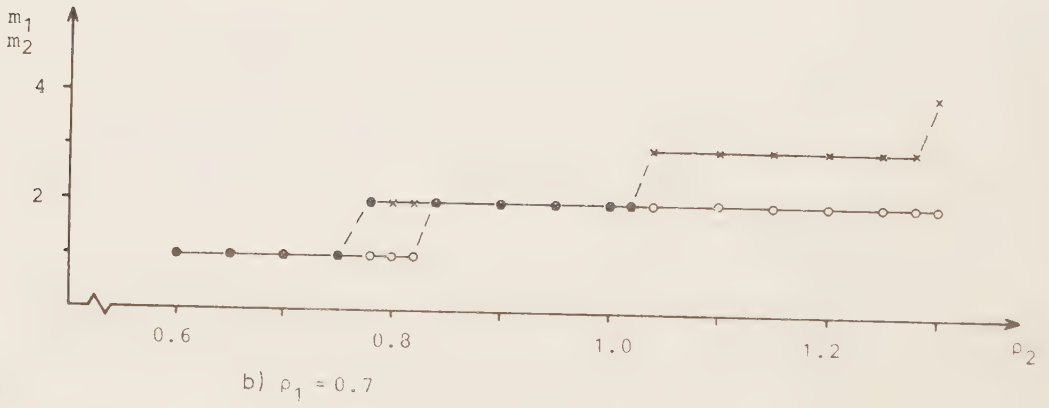
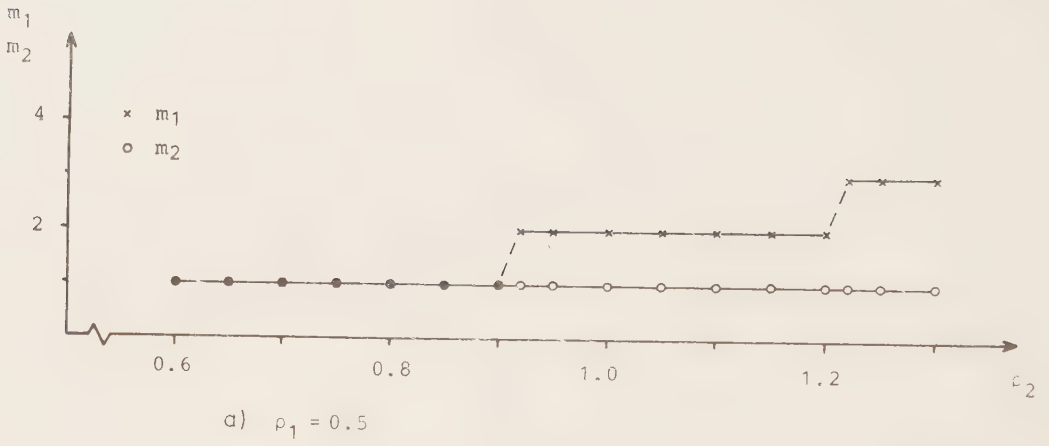


Figure 6.17 Unbalanced load

the SMA policy will come out more clearly under the unbalanced load situation presented in figure 6.17a to c. Take for instance figure 6.17c, where $\rho_1 = 0.9$ and ρ_2 varies from 0.56 to 1.30. In the interval $0.56 \leq \rho_2 \leq 0.66$ the SMA policy allocates 2 dedicated buffers to 2-packets and 1 dedicated buffer to 1-packets as $\rho_2 < \rho_1$. In the interval $0.68 \leq \rho_2 \leq 0.92$ both packet streams are allocated two buffers each and when ρ_2 increases further, the SMA policy allocates more dedicated buffers to 1-packets.

Tables 6.1 and 6.2 show the number of dedicated buffers under the fairly balanced and the unbalanced load situations respectively and when $n=3$, i.e. there are three outgoing transmission links. Here the SMA policy favours packets bound for the two least loaded transmission links. However, the SMA policy does not favour packets bound for the least loaded transmission link in comparison with the second least, but does protect them both.

As we will find in section 6.8 the allocation of dedicated buffers is in a more general case not only a function of offered load, but a function of the different arrival rates or the link capacities as well.

ρ_3	m_1	m_2	m_3
0.60	0	0	0
0.62	0	0	0
0.64	0	0	0
0.66	0	0	0
0.68	1	0	0
0.70	1	1	1
0.72	1	1	1
0.74	1	1	1
0.76	1	1	1
0.78	1	1	1
0.80	1	1	1
0.82	2	1	1
0.84	2	1	1
0.86	2	2	2
0.88	2	2	2
0.90	3	2	2
0.92	3	2	2
0.94	3	3	2
0.96	3	3	2
0.98	4	3	3
1.00	4	3	3
1.02	4	4	3
1.04	4	4	3
1.06	4	4	3
1.08	5	4	4
1.10	5	5	4
1.12	5	5	4
1.14	5	5	4
1.16	5	5	4
1.18	5	5	4
1.20	5	5	4
1.22	6	5	4
1.24	6	5	4
1.26	6	5	4
1.28	6	5	5

fairly balanced load; $\rho_2 = \rho_3 - 0.1$, $\rho_1 = \rho_3 - 0.2$

Table 6.1

ρ_3	m_1	m_2	m_3
0.54	1	0	1
0.56	1	0	1
0.58	1	1	1
0.60	1	1	1
0.62	1	1	1
0.64	1	1	1
0.66	1	1	1
0.68	1	1	1
0.70	1	1	1
0.72	1	1	1
0.74	1	1	1
0.76	1	1	1
0.78	1	1	1
0.80	1	2	1
0.82	1	2	1
0.84	1	2	1
0.86	1	2	1
0.88	1	2	1
0.90	1	2	1
0.92	2	3	2
0.94	2	3	2
0.96	2	3	2
0.98	2	3	2
1.00	2	4	2
1.02	2	4	2
1.04	2	4	2
1.06	3	4	2
1.08	3	4	2
1.10	3	5	2
1.12	3	5	2
1.14	3	5	2
1.16	3	5	2
1.18	3	5	2
1.20	3	6	2
1.22	3	6	2
1.24	4	6	2

unbalanced load; $\rho_1 = 0.5$, $\rho_2 = 0.7$

Table 6.2

Finally in tables 6.3 and 6.4 the number of dedicated buffers m_1 and m_2 are given as function of ρ_1 and ρ_2 where ρ_1 and ρ_2 vary in the interval 0 to 3.0.

[illegible]

Table 6.3 The number of dedicated buffers for 1-packets

ρ_1																																													
3.0	0	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5						
2.9	0	3	3	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5				
2.8	0	3	3	4	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.7	0	3	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	
2.6	0	3	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	
2.5	0	2	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	
2.4	0	2	3	4	4	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
2.3	0	2	3	4	4	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
2.2	0	2	3	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
2.1	0	2	3	3	4	4	5	5	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
2.0	0	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
1.9	0	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
1.8	0	2	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
1.7	0	2	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
1.6	0	1	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
1.5	0	1	2	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	4
1.4	0	1	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1.3	0	1	2	2	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1.2	0	1	1	2	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1.1	0	1	1	1	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
1.0	0	0	1	1	1	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
0.9	0	0	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
0.8	0	0	0	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
0.7	0	0	0	0	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
0.6	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.5	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.4	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0.0	0.5					1.0					1.5					2.0					2.5					3.0					ρ_2													

Table 6.4 The number of dedicated buffers for 2-packets

6.7.3 Cost reduction

In order to visualize the benefits in terms of lost packets per second obtained when using the SMA policy instead of the CS, we define a cost reduction function $f'(\underline{m})$ as

$$f'(\underline{m}') = \frac{f(\underline{0}) - f(\underline{m}')}{f(\underline{0})} \quad (6.44)$$

which describes the reduction of the number of lost packets per second, when using the SMA instead of the CS policy. Here $f(\underline{m})$ is our usual linear cost function

$$f(\underline{m}) = \sum_{i=1}^n \lambda_i B_i \quad (6.45)$$

The vector $\underline{m}'$ is the optimal buffer allocation vector derived by means of the SMA policy.

In figure 6.18 $f'(\underline{m}')$ is drawn under the unbalanced load situation when ρ_2 varies and with ρ_1 as a parameter ($\rho_1 = 0.5, 0.7, 0.8, 0.9$). We find $f'(\underline{m}')$ to have a local minimum, when the two loads are almost equal.

Figure 6.19 shows the cost reduction drawn under the unbalanced load situation when we have three outgoing transmission links. Here $\rho_1 = 0.5$, $\rho_2 = 0.7$ and ρ_3 is varied. We find the cost reduction reaching a value of around 0.4 under heavy load.

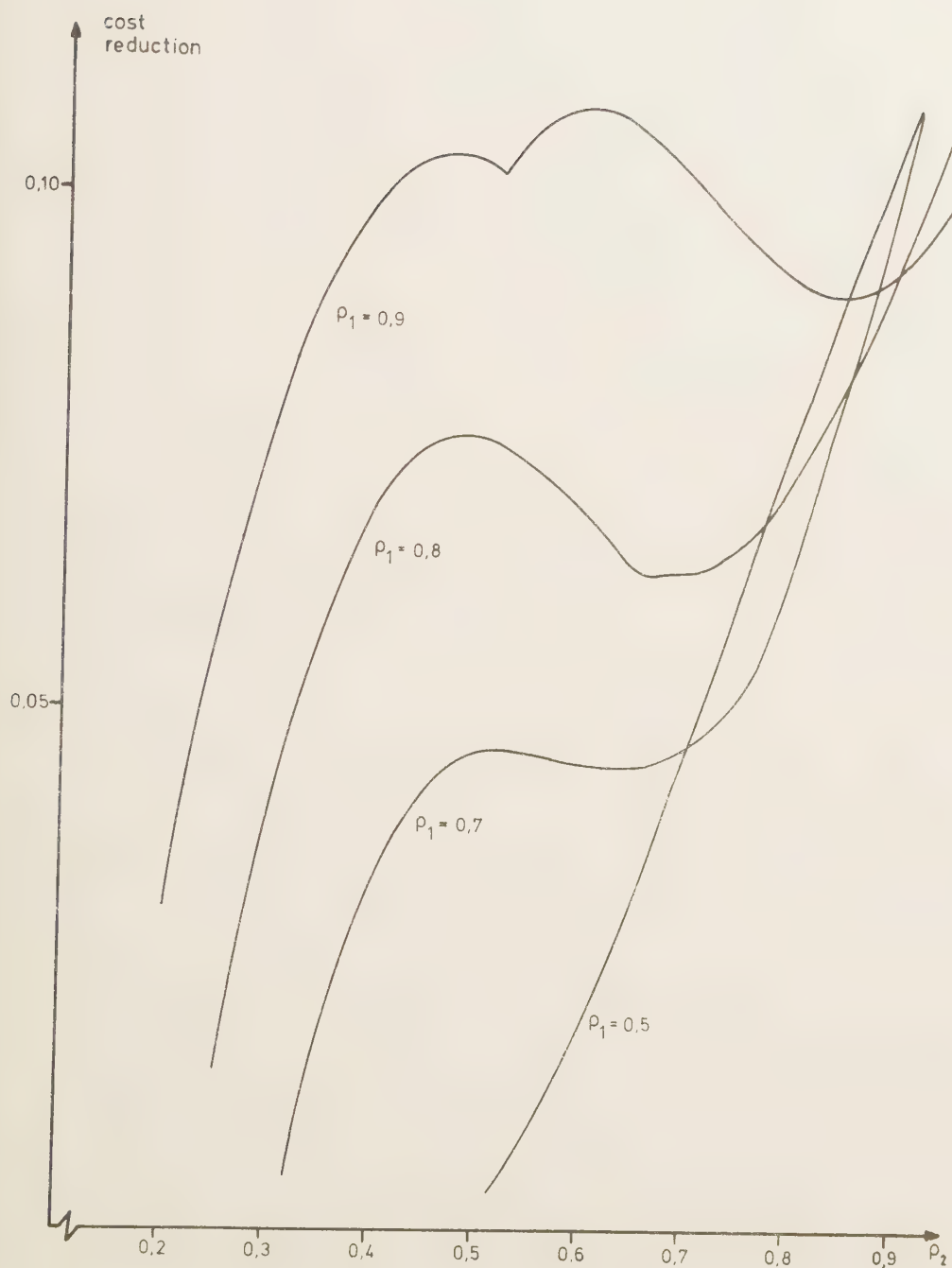


Figure 6.18 Unbalanced load

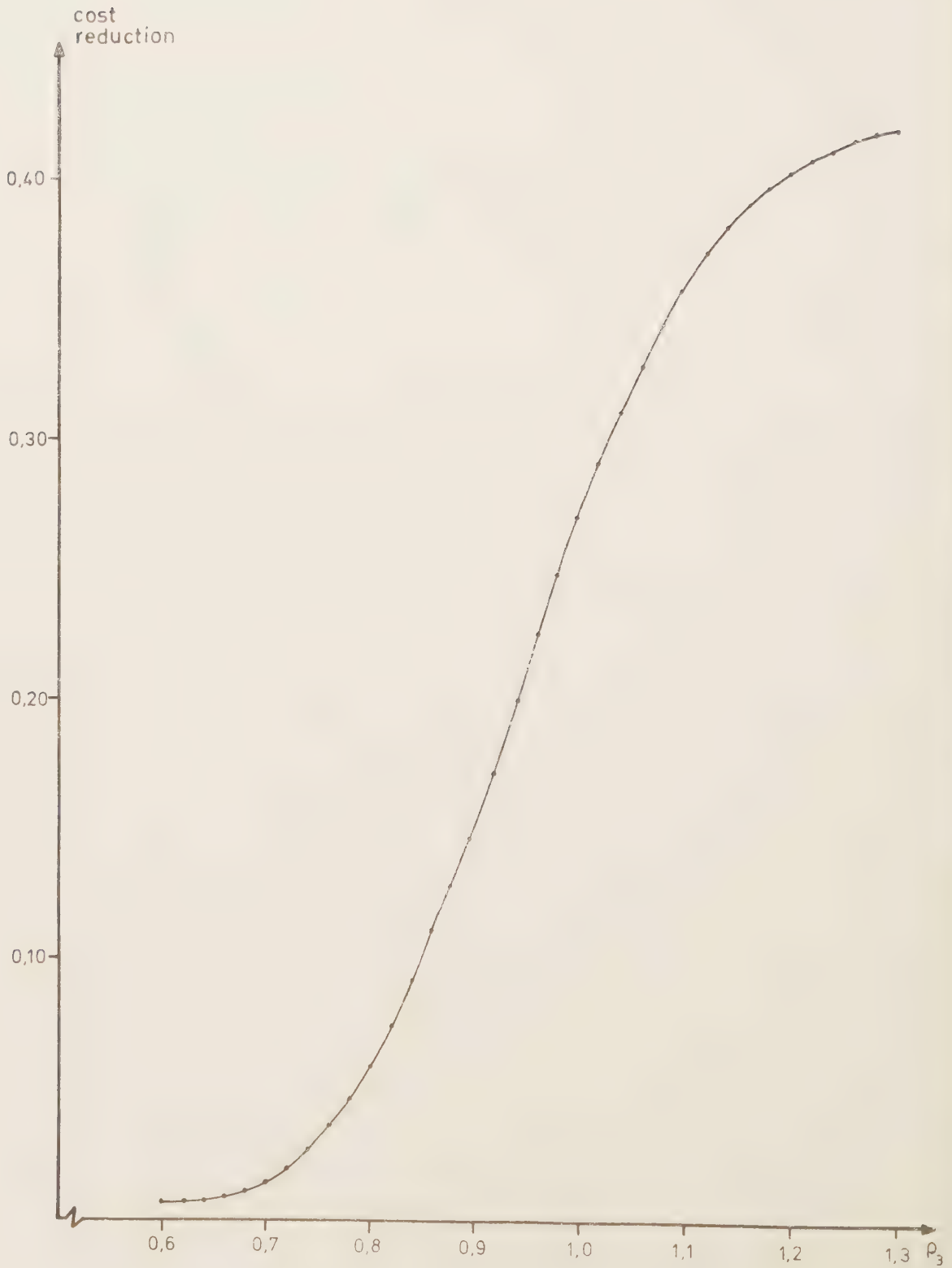


Figure 6.19 Unbalanced load, $\rho_1 = 9.5$, $\rho_2 = 0.7$

6.7.4 Throughput

If we use the linear cost function (6.13), just comprising the total number of lost packets per second, it is obvious that total throughput is maximized. Let us also observe the dependence between throughput and utilization. Throughput λ'_j on outgoing transmission link j

$$\lambda'_j = \lambda_j (1 - B_j) \quad (6.46)$$

and utilization U_j on link j

$$U_j = \rho_j (1 - B_j) \quad (6.47)$$

The corresponding total performance measures λ' and U are equal to the sum of the local measures taken over all transmission links

$$\lambda' = \sum_{j=1}^n \lambda'_j \quad (6.48)$$

$$U = \sum_{j=1}^n U_j \quad (6.49)$$

Thus if $C_1 = C_2 = \dots = C_n = C$

$$\lambda'_j = K \cdot U_j \quad (6.50)$$

and

$$\lambda = K \cdot U \quad (6.51)$$

where

$$K = \mu C \quad (6.52)$$

We could consequently in this section talk about utilization, as the normalized throughput and vice versa.

The expressions (6.46) to (6.51) show the direct correspondence between blocking probabilities presented in 6.7.1, throughput and utilization.

In this section we present total utilization as well as utilization of the individual links as a function of offered load, with one of the loads being held constant while the other varies.

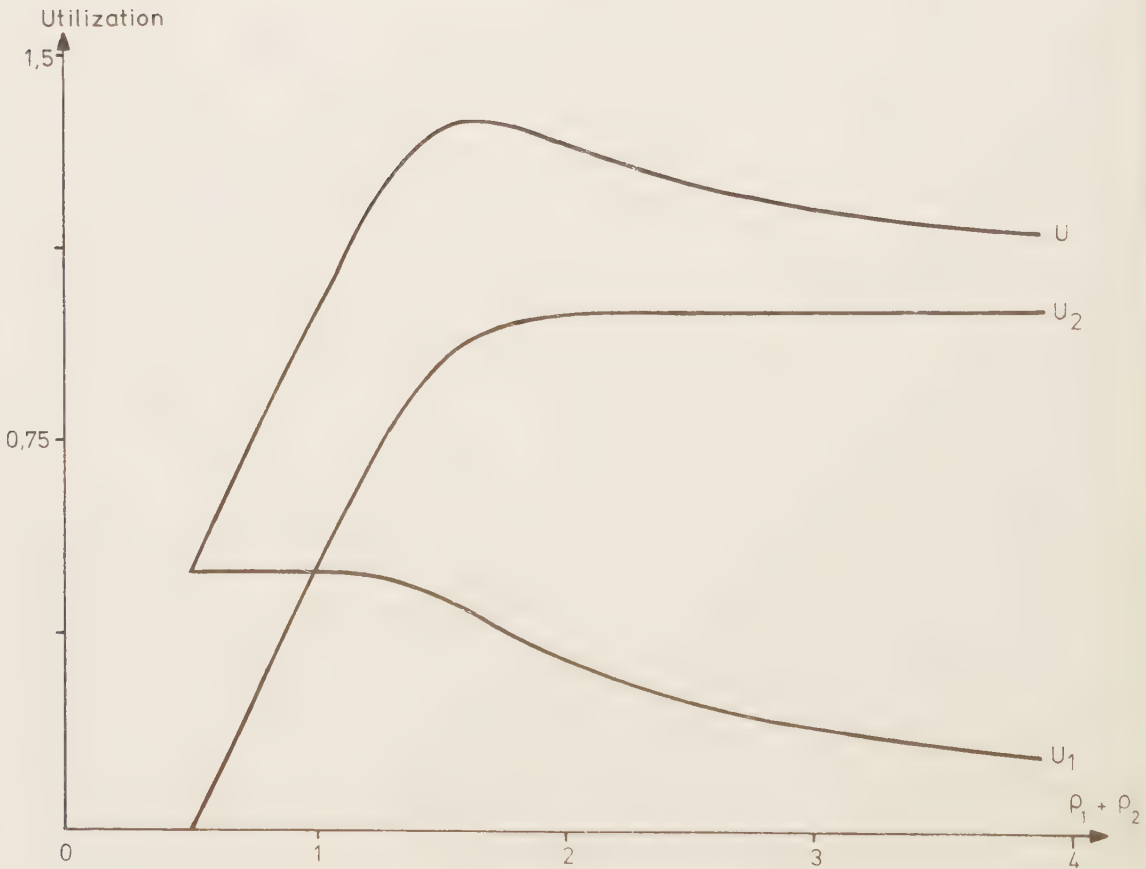


Figure 6.20 $\rho_1 = 0.5$



Figure 6.21 $\rho_1 = 0.9$

Figures 6.20 and 6.21 correspond to the CS policy and figures 6.22 and 6.23 to the SMA policy. In figure 6.20 and in 6.21 $\rho_1 = 0.9$ and ρ_2 is increased from 0.0. In both 6.20 and 6.21 we find utilization of transmission link 1 as well as total utilization being dramatically decreased.

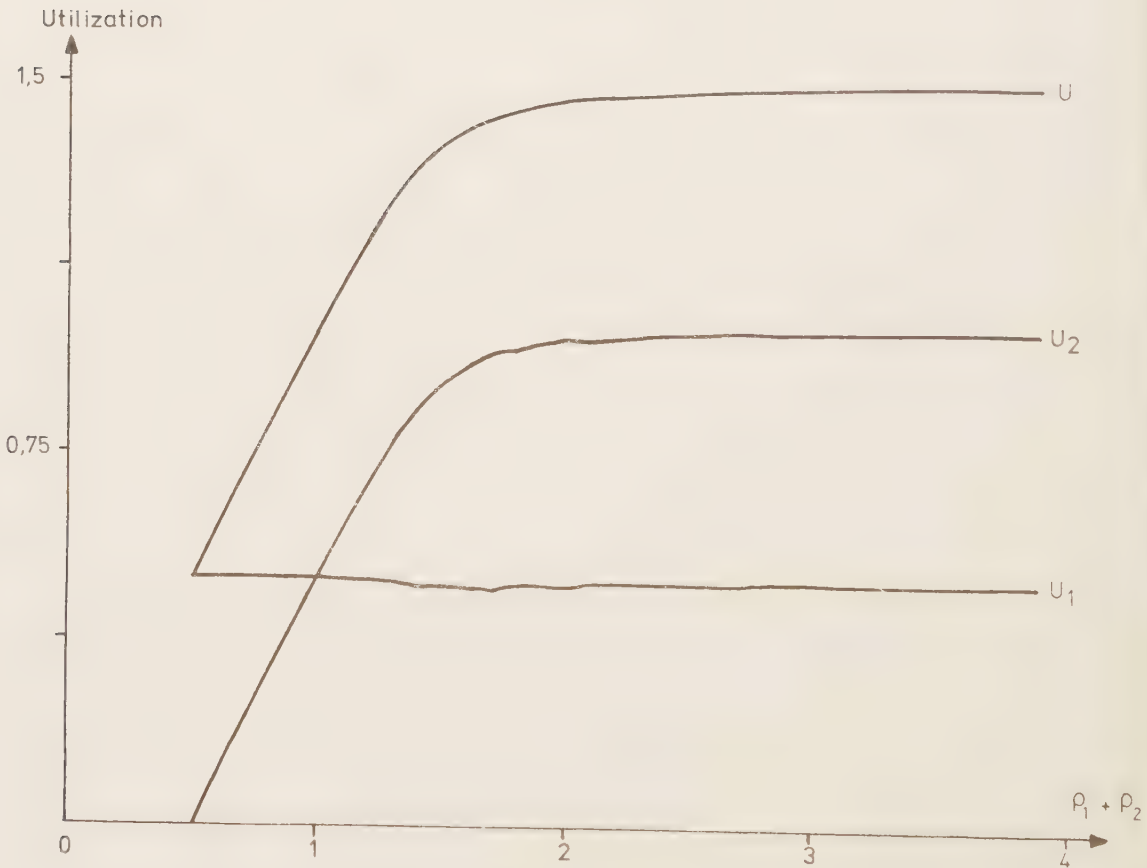


Figure 6.22 $\rho_1 = 0.5$

In the corresponding figures 6.22 and 6.23, based on the SMA policy, we find total utilization forming a non decreasing function of offered load.

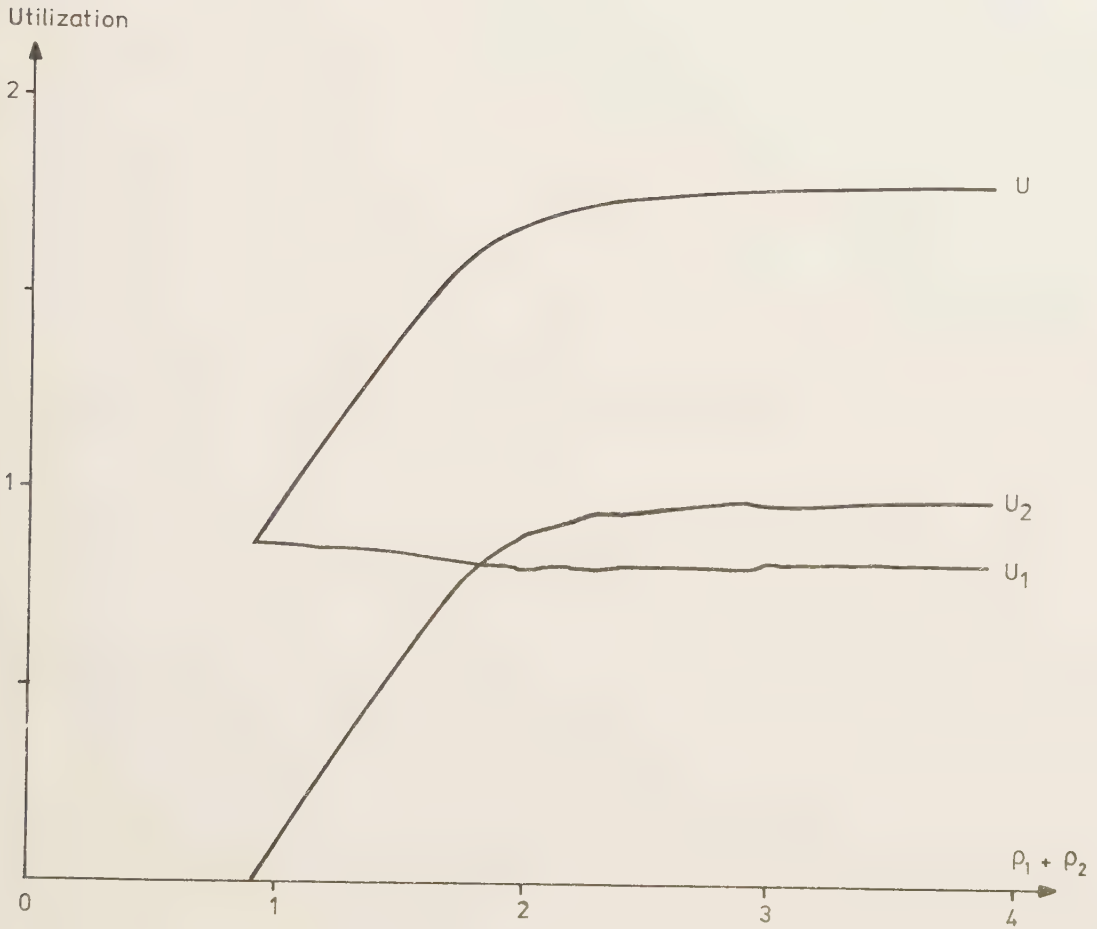


Figure 6.23 $\rho_1 = 0.9$

Irland [40] claims that his restricted policies, SMXQ and the square-root SMXQ, imply that total utilization forms a non decreasing function of load. However, as we will see in chapter 10, when we apply our policies to a complete net, total throughput may decrease with an increase of offered load, although only marginally.

In appendix B we compute the asymptotic values of utilization in both the SMA and the CS case. We find that

$$\lim_{\rho_2 \rightarrow \infty} U_1 = 0 \quad (6.53)$$

$$\lim_{\rho_2 \rightarrow \infty} U_2 = 1 \quad (6.54)$$

under the CS policy and that

$$\lim_{\rho_2 \rightarrow \infty} U_1 = \frac{\rho_1 (1 - \rho_1^{m_1})}{(1 - \rho_1^{m_1+1})} \quad (6.55)$$

$$\lim_{\rho_2 \rightarrow \infty} U_2 = 1 \quad (6.56)$$

under the SMA policy.

The limiting values in (6.53) and (6.54) are interesting and reinforce our knowledge that a huge traffic tends to hog most of the buffers. The expression in (6.55) corresponds to a M/M/1 system with exactly m_1 buffers. The m_0 common buffers are all, and indeed all, hogged by 2-packets.

6.7.5 Delays

Average delay for packets, i.e. time spent in the buffer system and transmission time, could be obtained by using Little's formula $\lambda'T = \bar{N}$, where $\bar{N}$ is the mean number of packets in the buffer system and λ' is the effective

arrival rate $\lambda(1-B)$. We present results based on both our fairly balanced as well as our unbalanced load situations. Furthermore we study both T_i , i.e. delays for i -packets, and T , i.e. delay for any packets. We restrict our presentation to the case with two outgoing transmission links.

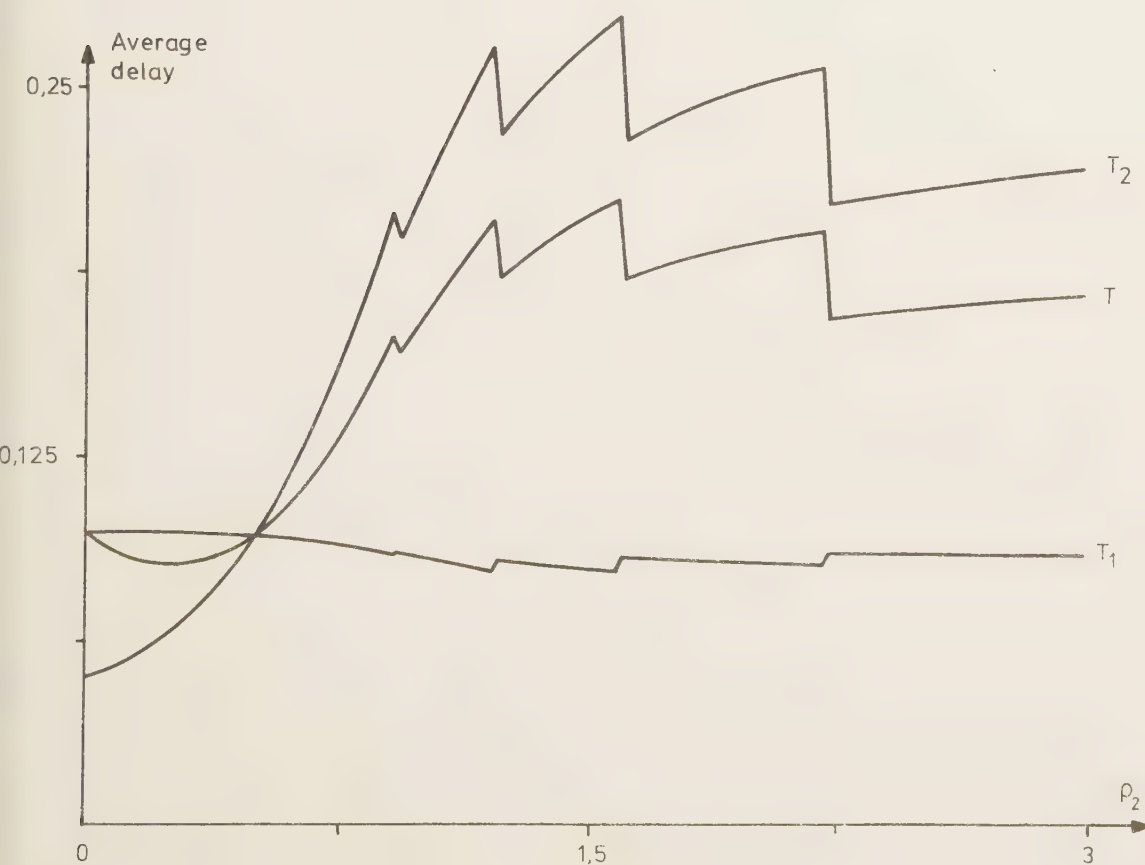


Figure 6.24 Unbalanced load, $\rho_1 = 0.5$

Figure 6.24 shows T_1 , T_2 and T under the SMA policy as a function of ρ_2 where $\rho_1 = 0.5$. Let us first discuss the behaviour of T in the interval $0 \leq \rho_2 \leq 0.5$. In the entire interval $m_1 = m_2 = 0$ and when $\rho_2 = 0$ $T_2 = 0.05$ since $\bar{x} = 1/\mu C = 0.05$. Moreover, when $\rho_2 = 0$ $T = T_1$ and when $\rho_2 = 0.5$ $T = T_1 = T_2$ and as T_1 decreases as long as 1-packets are not allocated any buffers, we obtain the appearance shown in figure 6.24. Figure 6.25 shows the same principle behaviour of delays, but here $\rho_1 = 0.7$.

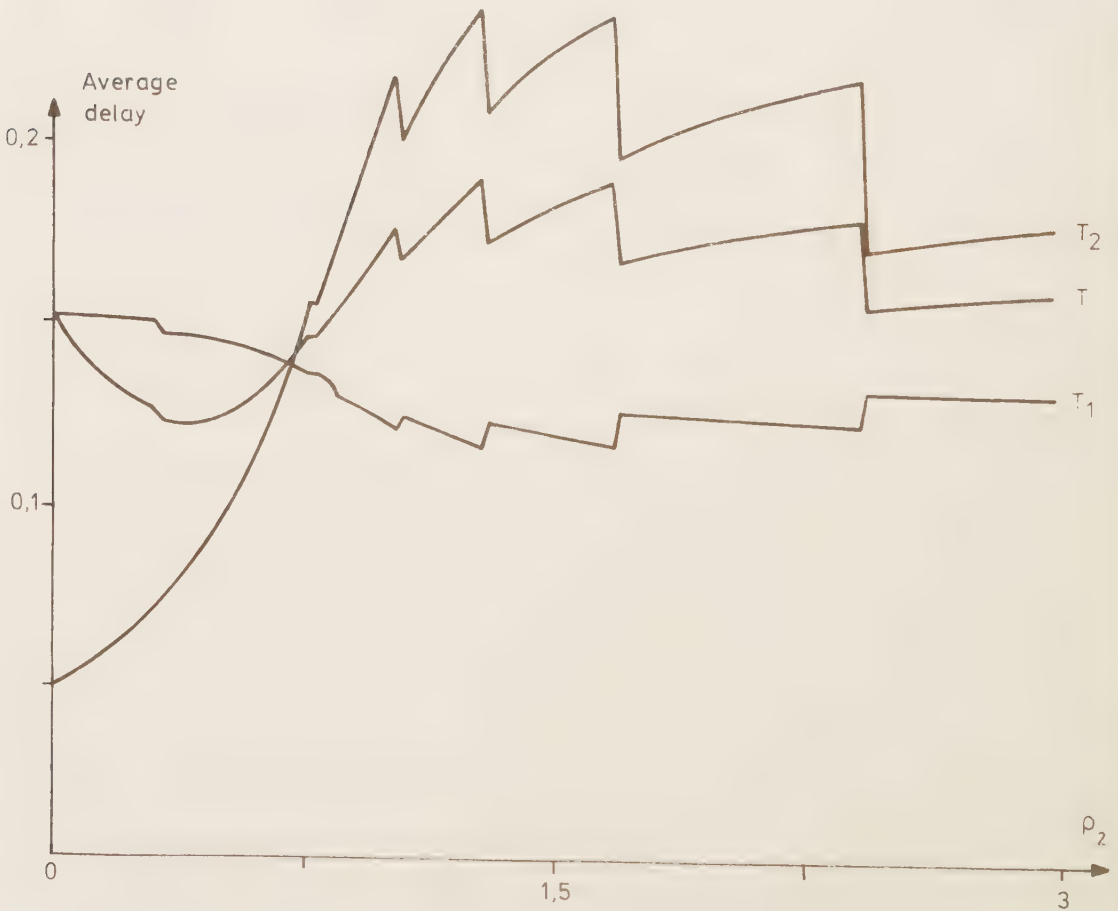


Figure 6.25 Unbalanced load, $\rho_1 = 0.7$

Figure 6.26 shows T as a function of ρ_2 under the CS, CP and SMA policies. Here we find the CS policy behaving poorly under heavy load and the CP policy better than the SMA policy. However, the number of dedicated buffers under the SMA policy are obtained when minimizing the number of lost packets per second and not when minimizing delay. The leaps in the SMA curve correspond to an increase of m_1 from 0 to 5 buffers and $m_2 = 1$ in the interval $0.75 < \rho_2 < 3$. The two curves (SMA and CP) seem to coincide in the two intervals $1.6 \leq \rho_2 \leq 2.0$ and $2.2 \leq \rho_2 \leq 3.0$. However, delays corresponding to the CP policy are a little bit shorter than those corresponding to the SMA policy. Under extreme load we know that the SMA policy will allocate buffers according to the CP policy.

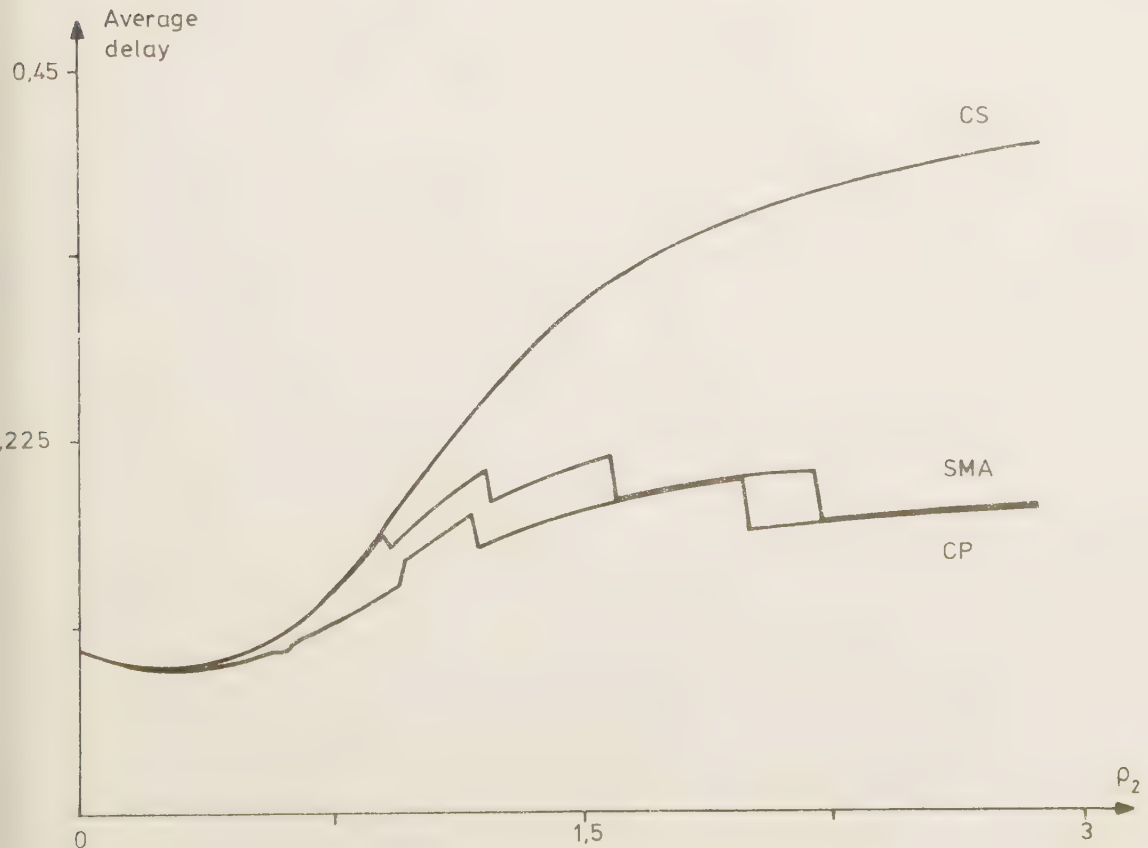


Figure 6.26 Unbalanced load, $\rho_1 = 0.5$

Figure 6.27 shows T as a function of ρ_2 for the CS, CP and SMA policies under the fairly balanced load situation. Again we find the CS policy behaving not as bad (but still bad) under the fairly balanced load situation as under the unbalanced load situation.

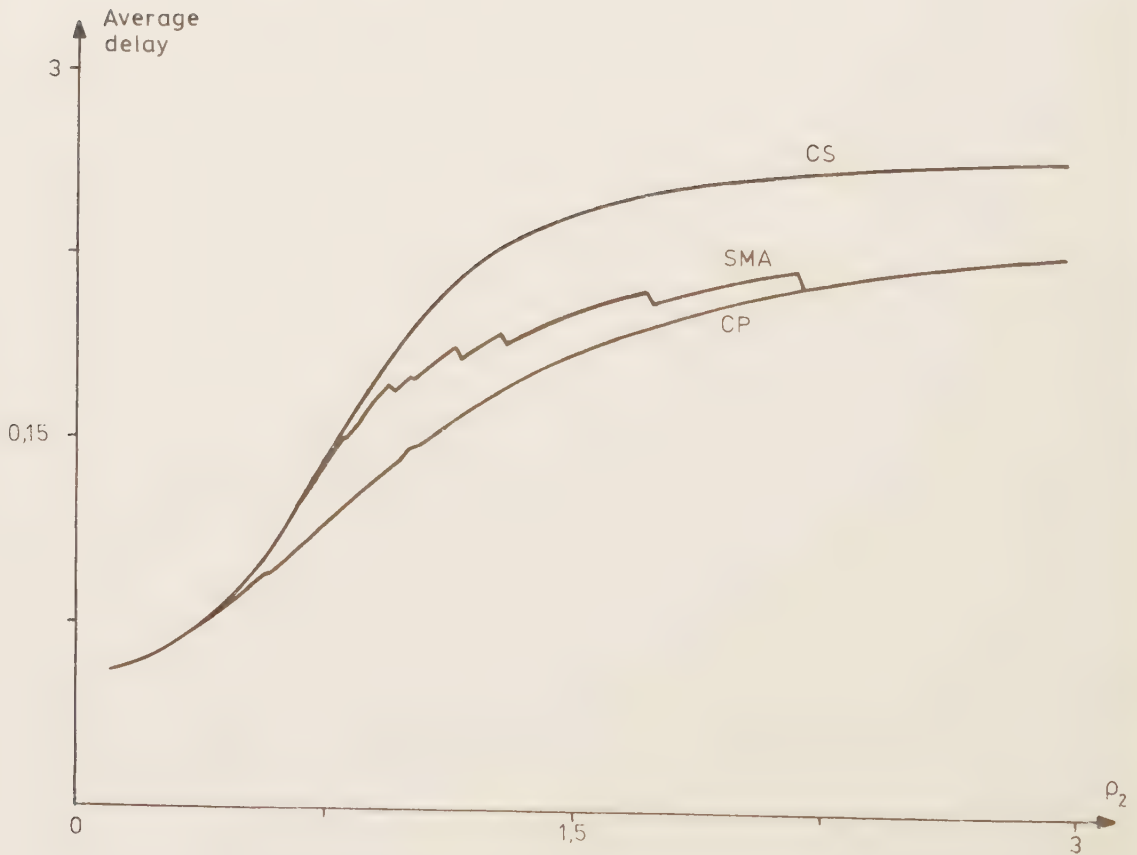


Figure 6.27 Fairly balanced load, $\rho_1 = \rho_2 - 0.1$

6.7.6 FSMA versus CS

In this section we compare the FSMA policy with the CS policy with respect to the number of blocked packets per second. In section 6.5.3 we introduced cost reduction defined as the decrease of the proportion between the SMA and CS policies of the number of lost packets per second. When comparing the FSMA and the CS policy we use the same definition of cost reduction. In this case, however, cost reduction may be negative and should in that case be seen as a cost increase.

In figure 6.28 we have drawn cost reduction under both the fairly balanced and the unbalanced load situations, when we have three outgoing transmission links. In both cases we have fixed the number of dedicated buffers to be unity for all packet types, i.e. $m_1 = 1$, $m_2 = 1$ and $m_3 = 1$ ($M = 20$). In the fairly balanced load situation $\rho_1 = \rho_3 - 0.2$, $\rho_2 = \rho_3 - 0.1$ and FSMA is identical to SMA within the interval $0.7 \leq \rho_3 \leq 0.82$, i.e. the SMA policy allocates one dedicated buffer to each of the packet streams in this interval. We find the FSMA policy better than the CS policy for $\rho_3 \geq 0.70$.

In the unbalanced load situation, where $\rho_1 = 0.5$ and $\rho_2 = 0.7$, the FSMA policy is identical to the SMA policy within $0.58 \leq \rho_3 \leq 0.78$. Here we find the cost reduction being positive when $\rho_3 \geq 0.42$.

One should be careful when comparing the two curves, as total load is not the same in the two cases. When ρ_3 is small, total load is bigger in the unbalanced than in the fairly balanced case and smaller when ρ_3 is big. Thus these curves are not sufficient to draw the conclusion that FSMA yields a better performance than CS in a larger interval under the unbalanced

load situation than in the fairly balanced. However, as seen in the previous sections, the CS policy performs poorly under the unbalanced load situation.

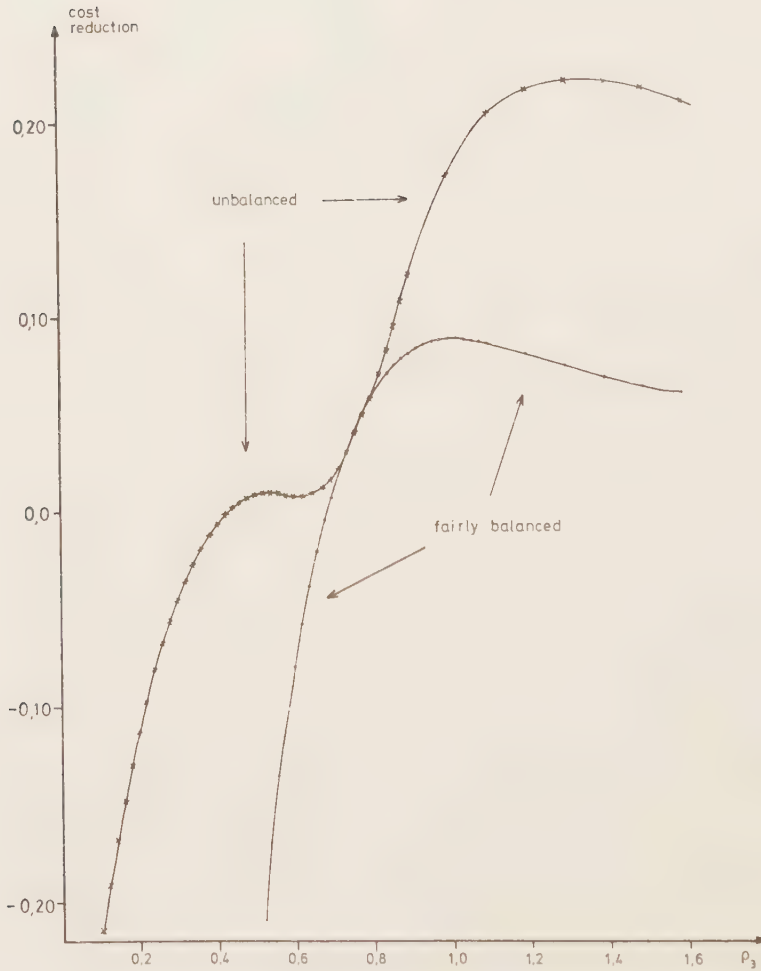


Figure 6.28

6.7.7 Utilization and Maximum Queue lengths

In [42] Irland treats in particular the ESMXQ policy and the Square root policy. In the ESMXQ policy we claim all b_i 's, i.e. the maximum number of i -packets in the buffers, to be equal $\forall i = 1, 2, \dots, n$. In chapter 5 we found the SMXQ and the SMA to be equal when $n=2$, that is when we have two outgoing transmission links.

In this section we investigate the differences between the ESMXQ and the SMXQ policies and as we here only present results where $n=2$, we prefer to call them ESMA and SMA respectively.

Our earlier presented results may indicate, as the SMA policy allocates more dedicated buffers to packets bound for a lightly loaded transmission link, that the ESMA policy should behave much worse than the SMA policy. However, though our results presented in this section do show differences, these differences are not that large.

We compare the two policies with respect to utilization or throughput and we do also compare the two optimal vectors $\underline{m}_g$ and $\underline{m}_I$ containing the number of dedicated buffers in the two cases (g: general, SMA, I: Irland, ESMA).

Figure 6.29 shows utilization or normalized throughput under the unbalanced load situation as a function of ρ_2 and when $\rho_1 = 0.9$ and $M = 10$. The figure shows the total, as well as the individual, utilization of the two transmission links. Utilization corresponding to the SMA policy is indicated with a g (general) and that to the ESMA policy with an I (Irland).

As seen in the figure, total utilization forms for both policies a nondecreasing function of offered load. However, the total utilization is higher under the SMA policy than under the ESMA policy. Furthermore we find the SMA policy to protect 1-packets better than the ESMA policy and it is this effect that contributes to the differences in total utilization.

If we take an example with a lower ρ_1 these effects will decrease.

In chapter 10 where we apply our restricted buffer sharing policies to a complete net we will see larger differences between the two policies.

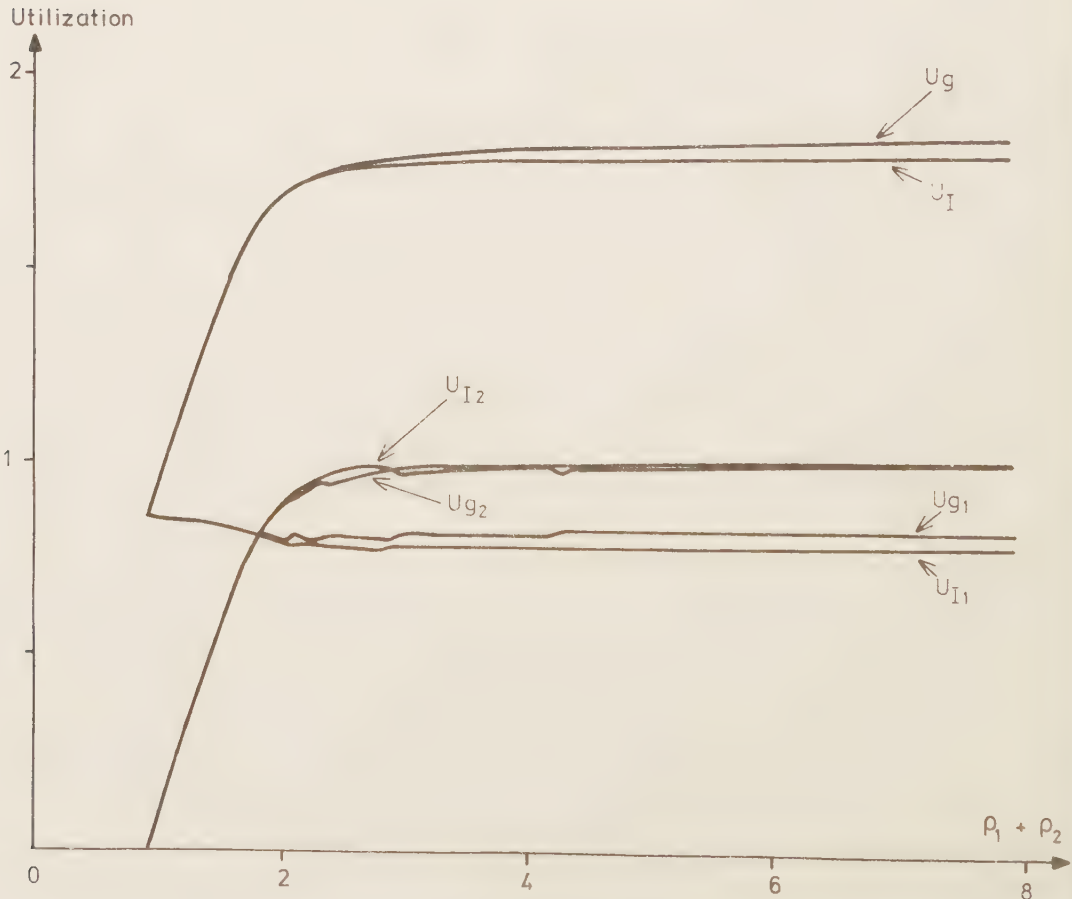


Figure 6.29 Unbalanced load, $\rho_1 = 0.9$

Table 6.5 shows the optimal $\underline{m}_I$ ($m_1 = m_2 = m_I$) as a function of ρ_1 and ρ_2 when $M = 10$. This table should be compared with table 6.6, which shows the optimal vector $\underline{m}_g$ (m_1, m_2). Comparing the two tables we find that the allocation of the number of dedicated buffers do differ significantly between the two policies. The differences between utilization is of course a result of the differences between the two tables.

The two tables could of course be translated to show the parameters under the ESMXQ and the SMXQ policies by setting $b_{ij} = M - m_{ij}$.

ρ_1																																								
3.0	0	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5			
2.9	0	3	3	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.8	0	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.7	0	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.6	0	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.5	0	2	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.4	0	2	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.3	0	2	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.2	0	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.1	0	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
2.0	0	2	3	3	4	4	4	4	5	5	5	5	5	5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.9	0	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.8	0	2	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.7	0	2	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.6	0	1	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.5	0	1	2	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.4	0	1	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.3	0	1	2	2	2	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.2	0	1	1	2	2	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.1	0	1	1	1	2	2	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
1.0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.9	0	0	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.8	0	0	0	1	1	1	1	2	2	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.7	0	0	0	0	1	1	1	1	2	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.6	0	0	0	0	0	1	1	1	1	2	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.5	0	0	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.4	0	0	0	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
0.3	0	0	0	0	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3		
0.2	0	0	0	0	0	0	0	0	0	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	0.0	0.5					1.0					1.5					2.0					2.5					3.0					ρ_2								

Table 6.5 The number of dedicated buffers for 1-packets as well as for 2-packets, ESMA

ρ_1																													ρ_2
3.0	0	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	
2.9	0	3	3	4	5	5	6	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	
2.8	0	3	3	4	5	5	5	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	
2.7	0	3	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	
2.6	0	3	3	4	4	5	5	6	6	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	
2.5	0	2	3	4	4	5	5	6	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	4	
2.4	0	2	3	4	4	5	5	5	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	4	
2.3	0	2	3	4	4	5	5	5	6	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	4	
2.2	0	2	3	3	4	4	5	5	6	6	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
2.1	0	2	3	3	4	4	5	5	5	6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
2.0	0	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
1.9	0	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
1.8	0	2	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
1.7	0	2	2	3	3	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
1.6	0	1	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
1.5	0	1	2	2	3	3	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4	
1.4	0	1	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
1.3	0	1	2	2	2	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
1.2	0	1	1	2	2	2	3	3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
1.1	0	1	1	1	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
1.0	0	0	1	1	1	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
0.9	0	0	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
0.8	0	0	0	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
0.7	0	0	0	0	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
0.6	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
0.5	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
0.4	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
0.3	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0.0	0.0	0.5	1.0	1.5	2.0	2.5	3.0																						

Table 6.6a The number of dedicated buffers for
i-packets, SMA

ρ_2																																	ρ_2		
3.0	0	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	4	4	4	4	5	5	5	5	5					
2.9	0	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.8	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.7	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.6	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.5	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.4	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.3	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.2	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.1	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
2.0	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.9	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.8	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.7	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.6	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.5	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.4	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.3	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.2	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.1	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
1.0	0	0	0	1	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.9	0	0	0	1	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.8	0	0	0	1	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.7	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.6	0	0	0	0	0	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.5	0	0	0	0	0	0	1	1	1	2	2	2	3	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.4	0	0	0	0	0	0	0	1	1	1	2	2	2	3	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.3	0	0	0	0	0	0	0	0	1	1	1	2	2	2	3	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.2	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3	3	4	4	4	4	4	4	4	4	4	5	5	5	5	5				
0.1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	2	3	3	3	3	3	3	3	3	3	4	4	4	4	4				
0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
	0.0	0.5					1.0					1.5					2.0					2.5					3.0								

Table 6.6b The number of dedicated buffers for
2-packets, SMA

6.8 The impact of offered load and arrival rate

In section 6.5 we found the blocking probabilities, B_i , being functions of the different m_j and the different ρ_j ($j = 1, 2, \dots, n$).

When minimizing the cost function with respect to the vector $\underline{m}$, the result, the optimal vector $\underline{m}_{\text{opt}}$, is a function of the different ρ_j and the different λ_j ($j = 1, 2, \dots, n$).

However, in the special case when all $C_i = C$, the optimal vector $\underline{m}_{\text{opt}}$ is just a function of the different ρ_j . This could be seen by multiplying the cost function with a constant $1/\mu C$. Therefrom

$$f(\underline{m})/\mu C = \sum_{i=1}^n \frac{\lambda_i}{\mu C} \cdot B_i; \quad (6.57)$$

$\Rightarrow$

$$f(\underline{m})/\mu C = \sum_{i=1}^n \rho_i B_i; \quad (6.58)$$

When $n=2$, i.e. there are two outgoing transmission links, the blocking probabilities are given by (6.10), (6.11) and (6.12). In this case the largest of the two loads tends to hog all the buffers, which makes the SMA policy allocate more dedicated buffers to the transmission link carrying the smallest load. This effect can be seen in figure 6.33 and in section 6.7.2. In

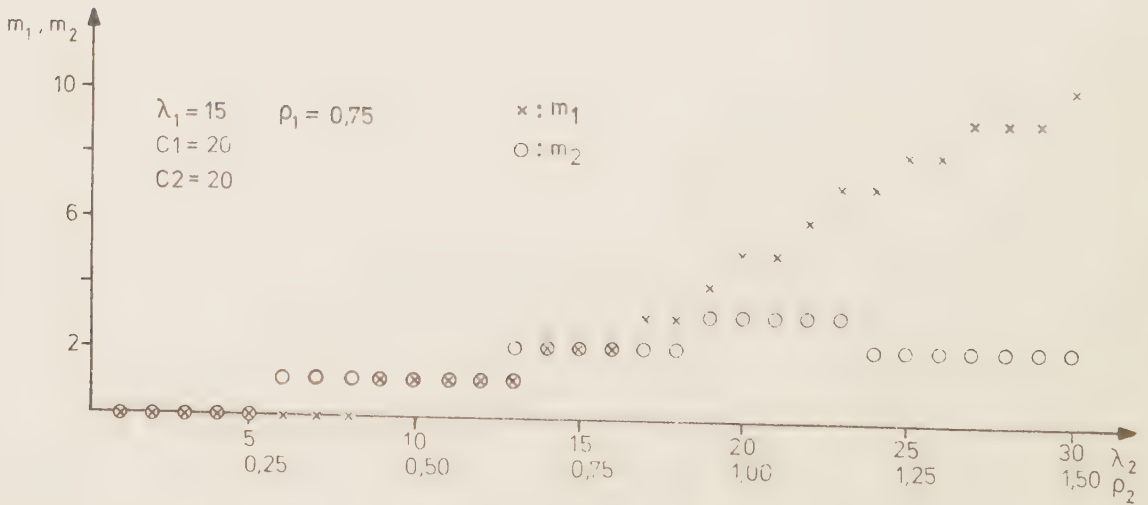
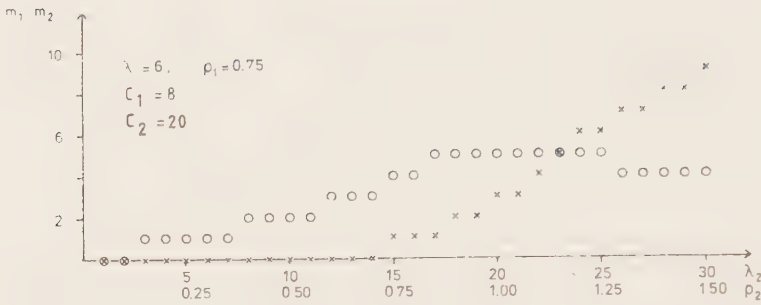


Figure 6.30

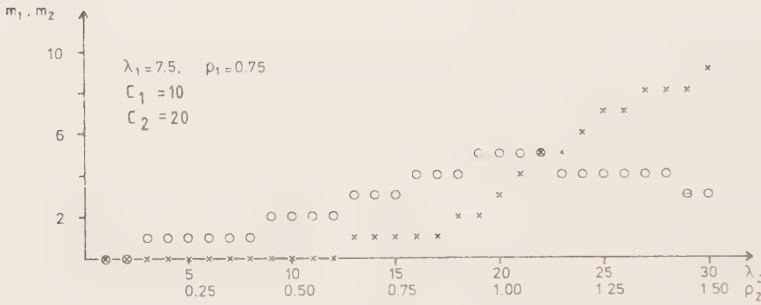
figure 6.30 $C_1 = C_2 = 20$ kbits per second and, as throughout this section, the mean packet length $1/\mu = 1000$ bits. Furthermore, $\lambda_1 = 15$ packets per second and thus $\rho_1 = 0.75$. In figure 6.30 λ_2 is increased from 1 to 30 packets per second, corresponding to an offered load of 0.05 to 1.50. As seen in the figure $m_1 \leq m_2$ for $\rho_1 \geq \rho_2$ and $m_1 \geq m_2$ for $\rho_1 \leq \rho_2$. In this example, as throughout this section (6.8), the total number of buffers, M , is equal to 20.

However, when $C_i \neq C_j$, the values of the different arrival rates contribute to the value of the optimal vector $\underline{m}_{\text{opt}}$. This effect could be seen in figures 6.31, 6.32 and 6.33. In each of these figures $C_2 = 20$ kbits per second and λ_2 is increased from 1 to 30 packets per second, corresponding to an offered load of 0.05 to 1.50. In figure 6.31 $\rho_1 = 0.75$, in figure 6.32 $\rho_1 = 0.5$ and in figure 6.33 $\rho_1 = 0.9$. In each of these three figures the parameter ρ_1 is built up by four different pairs of λ_1 and C_1 , and for each figure, this pair implies a constant ρ_1 . In figure 6.31a $\lambda_1 = 6$ packets per second and $C_1 = 8$ kbits per second. We see that the SMA policy favours 2-packets up to $\rho_2 = 1.15$ though $\rho_1 = 0.75$. Contrary to the case where $C_1 = C_2$, the SMA policy here allocates more dedicated buffers to 2-packets though the load on transmission link 2 is bigger than that on transmission link 1. This effect is of course rather obvious as we minimize the number of lost packets per second and as that measure is formed by multiplying the arrival rate by the function B_1 , consisting of the different loads.

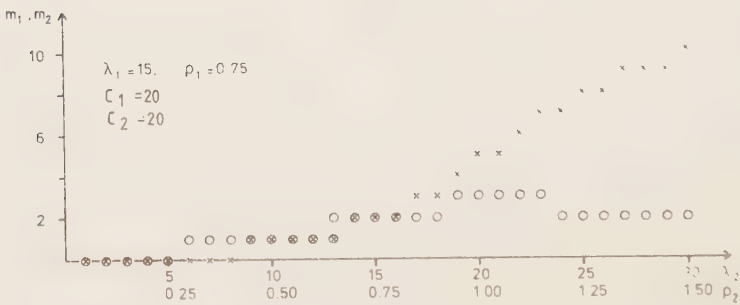
In figure 6.31a to d we find that the pattern of the number of dedicated buffers smoothly changes from favouring 2-packets to favouring 1-packets.



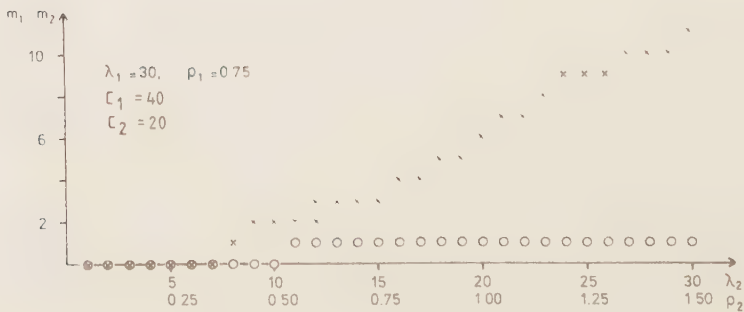
d)



f)



h)



d)

Figure 6.31

In figures 6.32 and 6.33 we find the same principle effects, but here for $\rho_1 = 0.5$ and $\rho_1 = 0.9$ respectively.

The allocation of buffers under the SMA policy when $C_i = C$ $\forall i = 1, 2, \dots, n$, i.e. allocate more buffers to packets bound for a less loaded transmission link, is not that straightforward when the number of transmission links is greater than 2. Take for instance a case with $n = 3$ $C_1 = C_2 = C_3 = C$ and $\rho_1 = 0.4$, $\rho_2 = 0.7$ and when ρ_3 is dramatically increased. Here the SMA policy protects 1-packets and 2-packets from the huge number of 3-packets. However, as a result of the minimization of the number of lost packets per second it is not a matter of course that 1-packets should be allocated more dedicated buffers than 2-packets though load on transmission link one is much smaller than that on link two. The SMA policy does protect the system as a whole and could sometimes favour especially 2-packets before 3-packets.

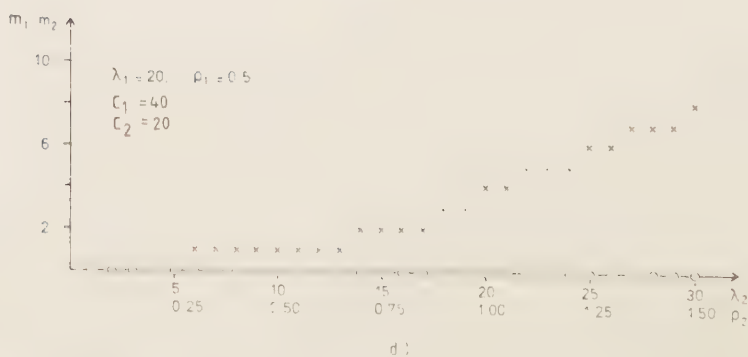
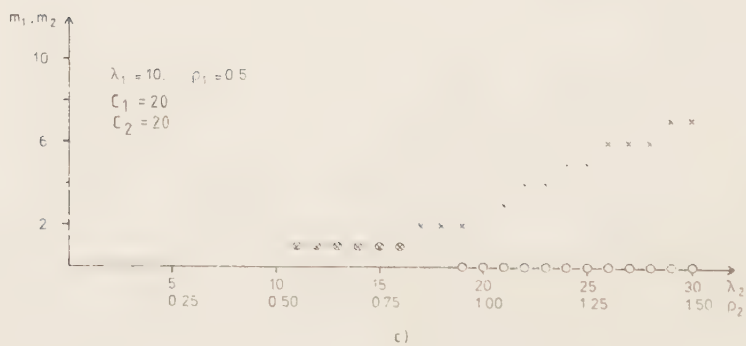
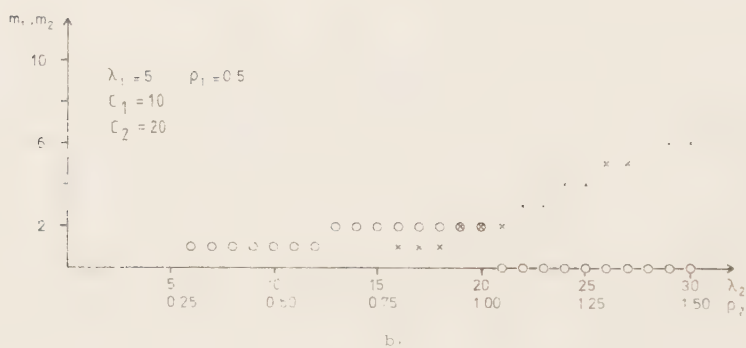
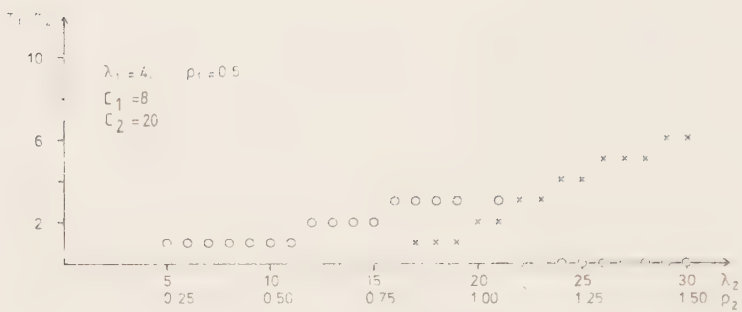
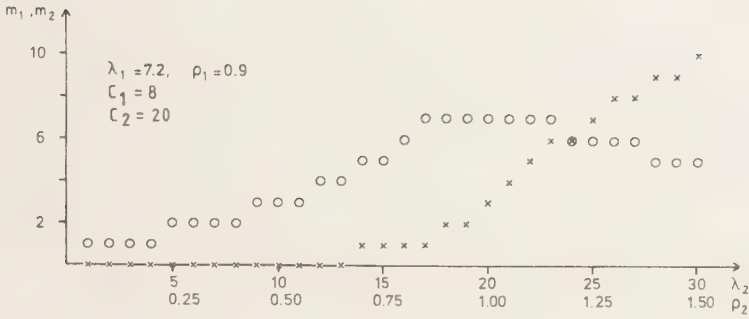
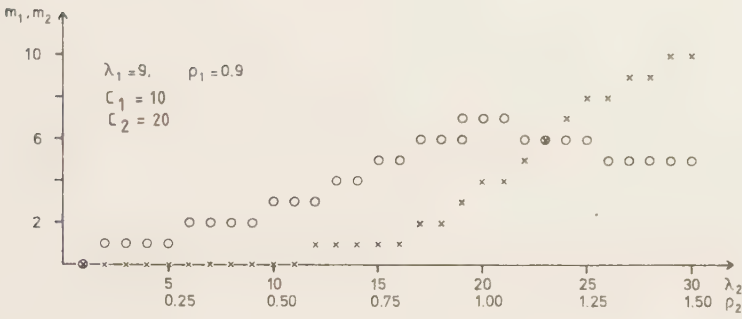


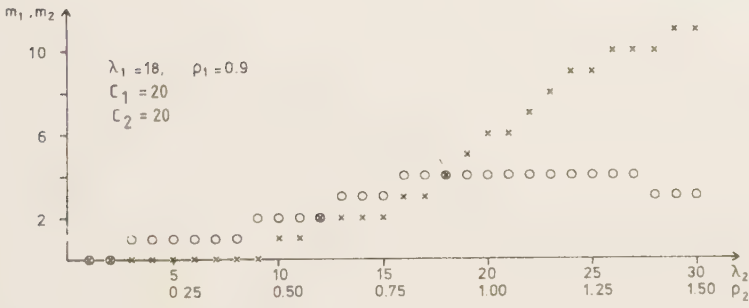
Figure 6.22



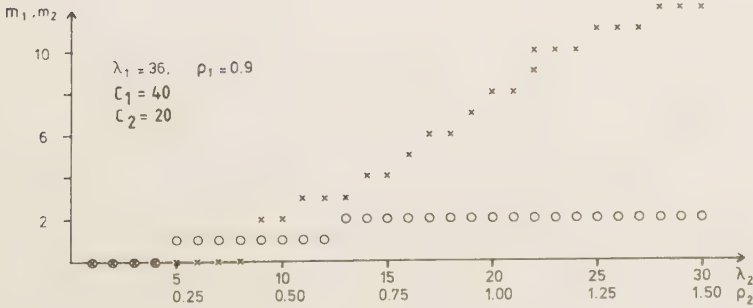
a)



b)



c)



d)

Figure 6.33

Figures 6.34 and 6.35 show the effects that appear in a node with three outgoing transmission links. In the two figures $C_1 = 100$, $C_2 = 50$ and $C_3 = 20$ kbits per second. In figure 6.34 the first two arrival rates are held constant, $\lambda_1 = 21$ and $\lambda_2 = 25$ packets per second, corresponding to $\rho_1 = 0.21$ and $\rho_2 = 0.70$ and λ_3 is increased from 5 to 19 packets per second, corresponding to $\rho_3 = 0.25, \dots, 0.95$

As seen in figure 6.34, although ρ_3 is very small ($\lambda_3 = 5 \Rightarrow \rho_3 = 0.25$), no dedicated buffers are allocated to 3-packets. This is so because λ_3 as well is small compared to λ_1 and λ_2 . Furthermore, 1-packets are not allocated more buffers than 2-packets, though $\rho_1 \ll \rho_2$ when $\lambda_1 < \lambda_2$. However, when ρ_3 increases, the SMA policy does favour 2-packets as $\rho_2 < \rho_3$ and $\lambda_2 > \lambda_3$.

In figure 6.35 we increase the three arrival rates simultaneously with one packet at a time. One must observe that this does not imply that offered load increases by the same factor for the three packet types. Here $\rho_1 = 0.21, 0.22, \dots, 0.45$; $\rho_2 = 0.50, 0.52, \dots, 0.98$; and $\rho_3 = 0.55, 0.60, \dots, 1.75$. The same effects that were found in figure 6.34 could be found here.

We have found that the SMA policy tends to allocate buffers in a higher degree to those transmission links, that are offered a smaller load and also to those, that are offered a larger arrival rate. These two effects do sometimes work in the same direction and sometimes they do not.

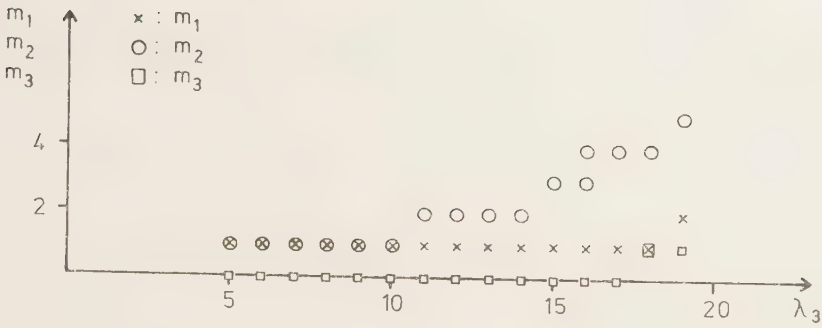


Figure 6.34

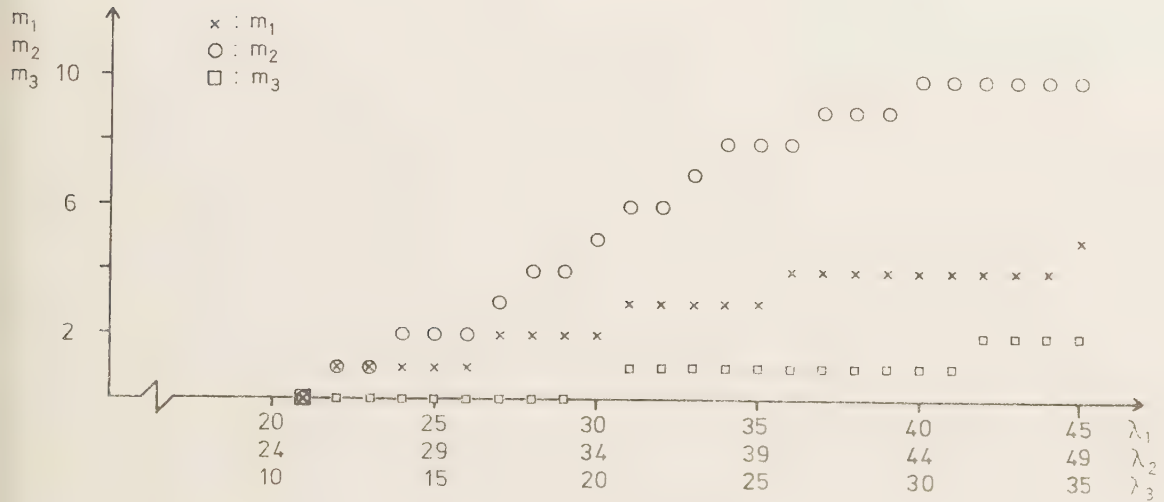


Figure 6.35

6.9 Conclusions

In this chapter we have implemented restricted buffer sharing policies in our packet switching node model. We have presented sample results in terms of blocking probabilities, throughput and delays under mainly the SMA, the CS and the CP policies. Within the SMA policy we have furthermore made comparisons between the ESMA, the FSMA policies and the three policies mentioned above. Furthermore the SMXQ policy and its relatives

the ESMXQ and the FSMXQ policies were treated under the assumption of two outgoing transmission links.

As the SMQMA policy (Sharing with Maximum Queues and Minimum Allocation) contains the other four pure restricted buffer sharing policies it must within this group be the best one. However, the benefits obtained when using the SMQMA policy instead of the best of the other four are in most cases utterly marginal.

The benefits and drawbacks of the CS and the CP policies in comparison with the SMA and the SMXQ policies are discussed rather thoroughly in section 6.7, (Sample results), but what about the benefits and drawbacks of the SMXQ policy in comparison with the SMA policy. In general the SMA policy yield better performance measures in terms of individual as well as total blocking probabilities and utilization under an unbalanced load situation. Under a moderate load and a fairly balanced load situation the SMXQ policy could be preferable.

During the last ten years much effort has been made in modelling packet switching networks in general and packet switching nodes in particular.

We have earlier, for instance in chapter 5, mentioned the, regarding restricted buffer sharing policies, pioneering efforts made by Kamoun [45] and Irland [40]. Rich and Schwartz [77] discuss a technique of sharing extra storage buffers among a number of output links in a single packet node. An article by Schweitzer and Lam [85] describes a more detailed model of a packet switch using the theory of networks-of-queues. In their model they do also take the effects of acknowledgement signals into account. Buttó, Colombo, Taggiasco and Tonietti [10]

deal with retransmission and time-outs. Schweitzer and Konheim [86] use an infinite capacity model with a limited number of customers in order to analyse finite buffer systems.

Chapter 7 - A RECURSIVE TECHNIQUE TO COMPUTE THE NORMALIZATION CONSTANT, THE BLOCKING PROBABILITIES AND THE MEAN NUMBER OF PACKETS

7.1 Introduction

As seen in the previous chapter, the expressions corresponding to the normalization constant, the different blocking probabilities and the mean number of packets tend to be rather huge and intractable when n , the number of outgoing transmission links, increases. Already when $n=2$, the different expressions are not only complex, but in addition we obtain special expressions for $\rho_1 \neq \rho_2 \neq 1$, $\rho_1 = \rho_2 = 1$, $\rho_1 \neq 1$ $\rho_2 = 1$ and $\rho_1 = 1$ $\rho_2 \neq 1$.

Furthermore, though we can state the different boundaries in the general n -dimensional case, it is hard, or even impossible, to state the expression for, say, the normalization constant.

Even if in principle we can state the different expressions, their complexity lend us to make mistakes, when we state them.

In this chapter we derive recursive techniques to compute the normalization constant, the blocking probabilities and the mean number of packets under the SMA policy. The same technique can of course be used under the other policies as well.

7.2 Feasible states within the n -dimensional state space

In the two-dimensional case, as we have seen earlier, it is easy both to state the boundaries and to depict

the state space. We claim that $k_j \geq 0$ ($j = 1, 2$), $k_1 + k_2 \leq M$ and that $k_j \leq m_j + m_0$ ($j = 1, 2$).

In general the state space Ω is an n -dimensional space with the following boundaries, of different dimensions. We call the boundaries "planes" independent of their dimensions in the n -dimensional space.

There are n planes for which

$$k_j = 0 \quad \forall j = 1, 2, \dots, n$$

Furthermore there are $\binom{n}{n} = 1$ planes of the form

$$\sum_{j=1}^n k_j = M$$

there are $\binom{n}{n-1} = n$ planes

$$\sum_{\substack{j=1 \\ j \neq \ell_1}}^n k_j = M - m_{\ell_1} \quad \forall \ell_1 = 1, 2, \dots, n;$$

there are $\binom{n}{n-2} = \frac{n(n-1)}{2}$ planes

$$\sum_{\substack{j=1 \\ j \neq \ell_1, \ell_2}}^n k_j = M - m_{\ell_1} - m_{\ell_2} \quad \begin{aligned} &\forall \ell_1 = 1, 2, \dots, n-1 \\ &\forall \ell_2 = \ell_1 + 1, \dots, n \end{aligned}$$

there are $\binom{n}{n-3} = \frac{n(n-1)(n-2)}{3!}$ planes

$$\sum_{\substack{j=1 \\ j \neq \ell_1, \ell_2, \ell_3}}^n k_j = M - m_{\ell_1} - m_{\ell_2} - m_{\ell_3} \quad \begin{aligned} &\forall \ell_1 = 1, 2, \dots, n-2 \\ &\forall \ell_2 = \ell_1 + 1, \dots, n-1 \\ &\forall \ell_3 = \ell_2 + 1, \dots, n \end{aligned}$$

.

.

.

In general there are $\binom{n}{n-k}$ planes

$$\begin{aligned} \sum_{j=1}^n k_j &= M - \sum_{i=1}^k m_i \ell_i & \forall \ell_i &= \ell_{i-1} + 1, \dots, n-k+1 \\ j \neq \ell_i & i=1, 2, \dots, k & \forall i &= 1, \dots, k \\ \ell_0 &= 0 \end{aligned}$$

.

.

.

there are $\binom{n}{1} = n$ planes

$$k_j = M - \sum_{\substack{i=1 \\ i \neq j}}^n m_i \quad \forall j = 1, 2, \dots, n$$

Take the 3-dimensional case as an example

First $k_1 = 0$, $k_2 = 0$, $k_3 = 0$

then $k_1 + k_2 + k_3 = M \quad \binom{3}{3} = 1$

$$k_1 + k_2 = M - m_3$$

$$k_1 + k_3 = M - m_2 \quad \binom{3}{2} = 3$$

$$k_2 + k_3 = M - m_1$$

$$k_1 = M - m_2 - m_3$$

$$k_2 = M - m_1 - m_3 \quad \binom{3}{1}$$

$$k_3 = M - m_1 - m_2$$

The space, that is restricted by these planes, is shown in figure 7.1.

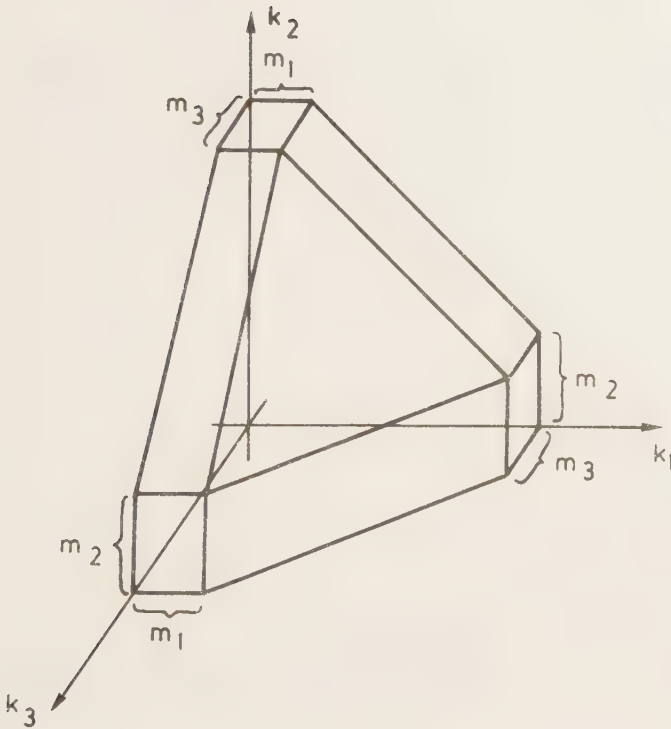


Figure 7.1

7.3 The normalization constant

Looking at the three dimensional state space, we can express $P(\underline{0})$ as a function of $P(\underline{0})$ in the two dimensional state space. The proposed technique sums up a two dimensional surface for each value of the third variable.

We illustrate this technique with an example. Let $M = 10$, $m_1 = 3$, $m_2 = 2$, $m_3 = 2$ and thus $m_0 = 3$. The three dimensional state space could be represented with $m_3 + m_0 + 1$ two dimensional state spaces.

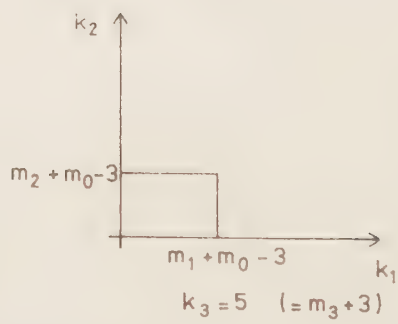
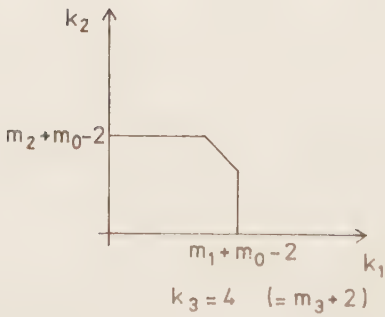
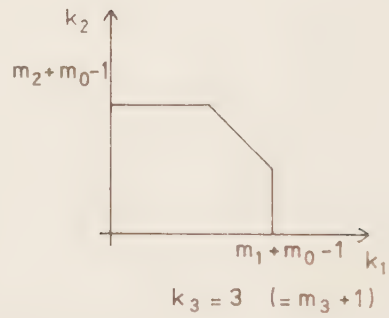
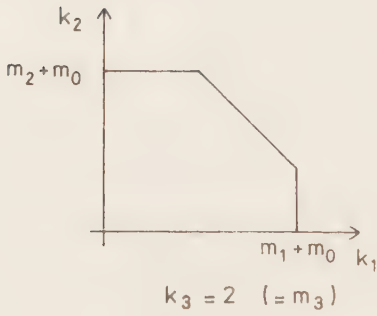
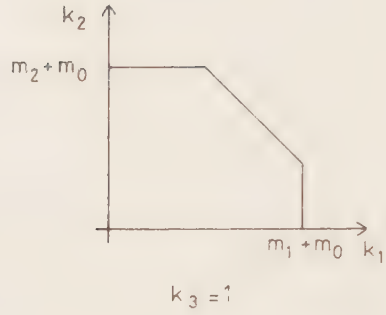
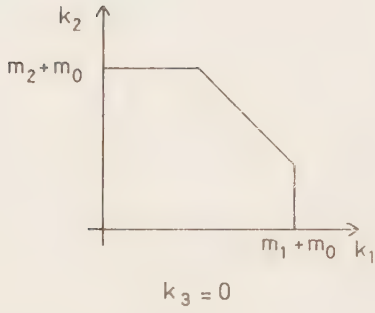


Figure 7.2

As seen in figure 7.2, the first three two dimensional state spaces correspond to the case when the number of non dedicated buffers equals m_0 and the other three to m_0-1 , m_0-2 and m_0-3 respectively.

Let

$$f_3(m_0) = \sum_{\Omega} \rho_1^{k_1} \rho_2^{k_2} \rho_3^{k_3} \quad (7.1)$$

where Ω is our three dimensional state space.

Let

$$f_2(k) = \sum_{\Omega_k} \rho_1^{k_1} \rho_2^{k_2} \quad (7.2)$$

where Ω_k is our two dimensional state space corresponding to k non allocated buffers.

Then

$$f_3(m_0) = P(0,0,0)^{-1} \quad (7.3)$$

and

$$f_3(m_0) = \sum_{k_3=0}^{m_3} \rho_3^{k_3} f_2(m_0) + \sum_{k_3=m_3+1}^{m_3+m_0} \rho_3^{k_3} f_2(m_0-k_3+m_3) \quad (7.4)$$

In general, $P(\underline{0})^{-1} = f_n(m_0)$

where

$$\begin{aligned} f_n(m_0) = & \sum_{k_n=0}^{m_n} \rho_n^{k_n} f_{n-1}(m_0) + \\ & + \sum_{k_n=m_n+1}^{m_n+m_0} \rho_n^{k_n} f_{n-1}(m_0-k_n+m_n); \end{aligned} \quad (7.5)$$

and

$$f_{n-1}(m) = \sum_{k_{n-1}=0}^{m_{n-1}} \rho_{n-1}^{k_{n-1}} f_{n-2}(m) + \\ + \sum_{k_{n-1}=m_{n-1}+1}^{m_{n-1}+m} \rho_{n-1}^{k_{n-1}} f_{n-2}(m-k_{n-1}+m_{n-1}); \quad (7.6)$$

Let $f_0(m) = 1 \quad \forall m: 0 \leq m \leq m_0$ which gives, as it should,

$$f_1(m_0) = \sum_{k_1=0}^{m_1} \rho_1^{k_1} \cdot 1 + \sum_{k_1=m_1+1}^{m_1+m_0} \rho_1^{k_1} \cdot 1 \quad (7.7)$$

$\Rightarrow$

$$f_1(m_0) = \sum_{k_1=0}^{m_1+m_0} \rho_1^{k_1}$$

$\Rightarrow$

$$f_1(m_0) = \frac{\rho_1^{m_1+m_0+1} - 1}{\rho_1 - 1} \quad (7.8)$$

As the first term in equation (7.5)

$$\sum_{k_n=0}^{m_n} \rho_n^{k_n} f_{n-1}(m_0) = f_{n-1}(m_0) \frac{\rho_n^{m_n+1} - 1}{\rho_n - 1} \quad (7.9)$$

we finally state

$$f_n(m_0) = f_{n-1}(m_0) \frac{\rho_n^{m_n+1} - 1}{\rho_n - 1} + \\ + \sum_{k_n=m_{n-1}+1}^{m_n+m_0} \rho_n^{k_n} f_{n-1}(m_0 - k_n + m_n); \quad (7.10)$$

$$f_0(m) = 1 \quad \forall m: 0 \leq m \leq m_0;$$

7.4 Blocking probabilities

In order to compute the different blocking probabilities we can use a technique similar to the one used to compute the normalization constant. For B_1 , the blocking probability for 1-packets, we have

$$B_1 = g_n(m_0) * P(\underline{0}) \quad (7.11)$$

where

$$g_n(m) = \sum_{k_n=0}^{m_n} \rho_n^{k_n} g_{n-1}(m) + \sum_{k_n=m_n+1}^{m_n+m} \rho_n^{k_n} g_{n-1}(m-k_n+m_n) \quad (7.12)$$

and

$$g_1(m) = \rho_1^{m_1+m} \quad \forall m: 0 \leq m \leq m_0 \quad (7.13)$$

If we rewrite the first term in (7.12) we obtain

$$g_n(m) = g_{n-1}(m) \frac{(1 - \rho_n^{m_n+1})}{(1 - \rho_n)} + \sum_{k_n=m_n+1}^{m_n+m} \rho_n^{k_n} g_{n-1}(m-k_n+m_n); \quad (7.14)$$

In this case it is harder to define $g_0(m)$. When $m > 0$ $g_0(m)$ ought to be zero but $g_0(0)$ is undefined.

The blocking probabilities corresponding to the other j -packets ($j = 2, 3, \dots, n$) can be obtained by renumbering the links.

7.5 The mean number of packets

The mean number of packets in the node is given by

$$\bar{N} = \sum_{\Omega} (k_1 + k_2 + \dots + k_n) P(k_1, k_2, \dots, k_n) \quad (7.15)$$

where

Ω is the state space.

In order to compute delays for j -packets in a node, we are interested in the marginal mean value for the number of j -packets in the node,

$$\bar{N}^j = \sum_{\Omega} k_j P(k_1, k_2, \dots, k_n) \quad (7.16)$$

If we renumber the different packet types, i.e. we call j -packets 1-packets and vice versa and define an auxiliary function $\bar{N}_n^1(m)$, where n indicates: n -dimensional and 1 : 1-packets, as

$$\bar{N}_n^1(m) = \sum_{k_n=0}^{m_n} \rho_n^{k_n} \bar{N}_{n-1}^1(m) + \sum_{k_n=m_n+1}^{m_n+m} \rho_n^{k_n} \bar{N}_{n-1}^1(m-k_n+m_n) \quad (7.17)$$

$$\bar{N}_1^1(m) = \sum_{k_1=0}^{m_1+m} k_1 \cdot \rho_1^{k_1} \quad (7.18)$$

Then

$$\bar{N}^1 = \bar{N}_n^1(m_0) \cdot P(\underline{0}) \quad (7.19)$$

We also observe that

$$\sum_{k=0}^P k \cdot \alpha^k = \alpha \sum_{k=0}^P k \cdot \alpha^{k-1}; \quad (7.20)$$

$$\alpha \sum_{k=0}^P k \cdot \alpha^{k-1} = \alpha \frac{\partial}{\partial \alpha} \left(\sum_{k=0}^P \alpha^k \right) \quad (7.21)$$

and

$$\sum_{k=0}^P \alpha^k = \frac{1 - \alpha^{P+1}}{1 - \alpha} \quad (7.22)$$

$$\frac{\partial}{\partial \alpha} \left(\frac{1 - \alpha^{P+1}}{1 - \alpha} \right) = \frac{(1 - \alpha^P (P + 1 - P \cdot \alpha))}{(1 - \alpha)^2} \quad (7.23)$$

Thus

$$\bar{N}_1^1(m) = \frac{(1 - \rho_1^{m_1+m} (m_1+m+1 - (m_1+m) \cdot \rho_1))}{(1 - \rho_1)^2} \quad (7.24)$$

Finally we state

$$\bar{N} = \sum_{j=1}^n \bar{N}^j \quad (7.25)$$

7.6 Conclusions

The described technique shows, in comparison with the directly computed formulae, benefits in terms of "more handable", "more surveyable" and "more general" (the directly computed formulae appear in different shapes when for instance $\rho_i = 1$, $\rho_i = \rho_j$).

The technique to solve the normalization constant, the blocking probabilities and the mean number of packets in a node has similarities with the technique to compute the normalization constant in queueing network theory. This technique was first introduced by Buzen [11] and later used by Irland [40]. However, the state space in their applications makes it possible to derive the normalization constant, the blocking probabilities and the mean number of packets in a node by means of the z-transform.

Chapter 8 - A PACKET SWITCHING NODE MODEL WITH ACKNOWLEDGEMENT SIGNALS AND RETRANSMISSION

8.1 Introduction

In our model of a packet switch in chapter 6 a packet was held in one of the buffers, first while waiting for the appropriate transmission link to become free and then a copy of the packet was held in the same buffer during the transmission. In this chapter we treat a model in which the packet is held not only during transmission, but until an acknowledgement signal is received (8.2), and if this acknowledgement is negative, until the transmission is successfully completed (8.3). The steady state probabilities, derived in appendix A, have product form, and thus the same fundamental structure as those found in chapter 6.

8.2 A model with acknowledgement signals

We now extend our model in chapter 6 with acknowledgement signals and, as in chapter 6, assume packets, bound for transmission link i (i -packets), to arrive to the node according to a Poisson-process with the arrival rate λ_i . The packet lengths are exponentially distributed with mean $\frac{1}{\mu}$. Immediately after completion of a transmission, an acknowledgement signal is sent back and the corresponding packet, stored in the buffer, is dropped when the signal is received. The length of an acknowledgement signal is also exponentially distributed, but with mean $\frac{1}{\nu}$. These assumptions imply that acknowledgement signals have priority over other signals. This could be motivated as follows: When an acknowledgement signal is to be sent, and the appropriate transmission link is busy, the acknowledgement signal is "piggybacked" on the transmitted packet. Further-

more we do not consider acknowledgement signals confirming transmissions towards our node.

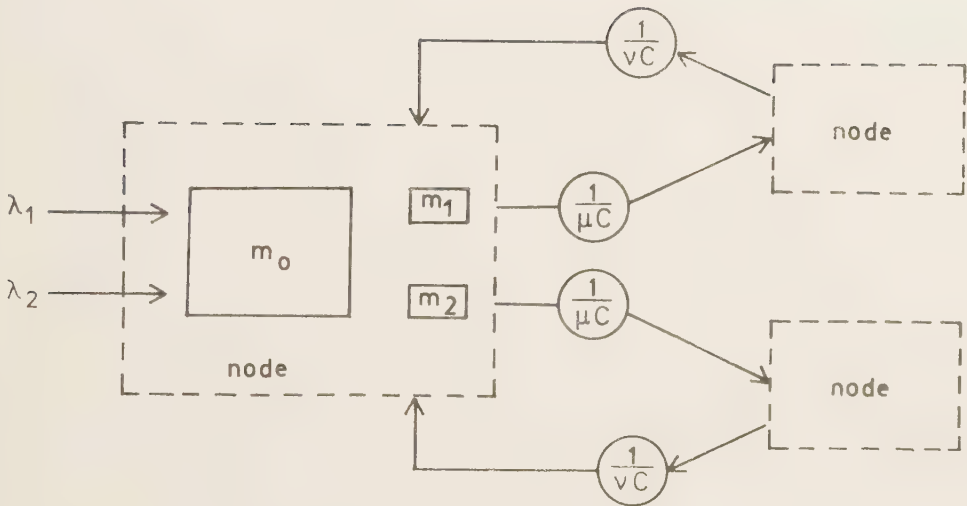


Figure 8.1

A packet, waiting in a buffer for transmission, enters the transmission link immediately after the link becomes free and thus does not have to wait for the just completed transmission to be acknowledged.

In this chapter two different buffer sharing policies are considered: Complete Sharing (CS) and Sharing with Minimum Allocation (SMA).

8.2.1 The steady state distribution

It is with the following assumptions

- i) Packets bound for output link i arrive according to a Poisson process with rate λ_i .
- ii) The packet lengths, as well as the lengths of the acknowledgement signals, are exponentially distributed with mean $\frac{1}{\mu}$ and $\frac{1}{\nu}$ respectively.
- iii) The number of buffers are limited and equals M .
- iv) The capacity of the outgoing transmission links are equal to C bits/sec.

possible to derive the steady state distribution (see appendix A).

$P(k_1, k_2, j_1, j_2) = P(k_i \text{ packets in the buffers waiting for transmission over link } i \text{ or for transmission over link } i \text{ to be completed, and } j_i \text{ packets in the buffers waiting for an acknowledge corresponding to link } i, i = 1, 2)$

which can be written

$$P(k_1, k_2, j_1, j_2) = \left(\frac{\lambda_1}{\mu C}\right)^{k_1} \left(\frac{\lambda_2}{\mu C}\right)^{k_2} \left(\frac{\lambda_1}{\nu C}\right)^{j_1} \left(\frac{\lambda_2}{\nu C}\right)^{j_2} P(0, 0, 0, 0)$$

We also have

$$P(k_1, k_2) = P(k_i \text{ packets corresponding to link } i \text{ in the buffers, } i = 1, 2)$$

$$P(k_1, k_2) = \frac{(\rho_{22}^{k_1+1} - \rho_{12}^{k_1+1})(\rho_{21}^{k_2+1} - \rho_{22}^{k_2+1})}{(\rho_{11} - \rho_{12})(\rho_{21} - \rho_{22})} P(0, 0, 0, 0)$$

where

$$\rho_{11} = \frac{\lambda_1}{\mu C}, \quad \rho_{12} = \frac{\lambda_1}{vC}, \quad \rho_{21} = \frac{\lambda_2}{\mu C} \text{ and } \rho_{22} = \frac{\lambda_2}{vC}$$

In general we find

$$P(\underline{k}, \underline{j}) = \prod_{i=1}^n \left(\frac{\lambda_i}{\mu C} \right)^{k_i} \left(\frac{\lambda_i}{vC} \right)^{j_i} \cdot P(\underline{0})$$

where

$$\underline{k} = (k_1, k_2, \dots, k_n) \text{ and } \underline{j} = (j_1, j_2, \dots, j_n)$$

and

$$P(\underline{k}) = P(k_i \text{ packets corresponding to link } i \text{ in the buffers } \forall i = 1, 2, \dots, n)$$

$$P(\underline{k}) = \prod_{i=1}^n \frac{(\rho_{i1}^{k_i+1} - \rho_{i2}^{k_i+1})}{(\rho_{i1} - \rho_{i2})} \cdot P(\underline{0})$$

8.2.2 Performance Measures

In order to obtain expressions for

- a) blocking probabilities
- b) utilization of the transmission links
- c) the number of dedicated buffers in the SMA case

we compute

- i) $P(\underline{0}) = P(0, 0) = P(\text{no packets in the buffers})$
- ii) $B_i = P(i\text{-packet being blocked}).$

The normalization constant $P(0)$ as well as the blocking probabilities and the mean number of packets could, in the general n -dimensional case, be derived by means of a recursive technique like the one presented in chapter 7. We focus our derivation on the case $n=2$.

Let

$$f_1(x, y) = \frac{x \cdot y}{(1-y)} \left(\frac{(1-x)^{M+1}}{(1-x)} - \frac{y(x^{M+1} - y^{M+1})}{(x-y)} \right);$$

$$f_2(x) = \frac{x(1-x^{m_1+1})}{(1-x)};$$

$$f_3(x) = \frac{x(1-x^{m_2+m_0+1})}{(1-x)};$$

$$f_4(x, y) = \frac{y \cdot x^{m_1+2}}{(1-y)} \left(\frac{(1-x^{m_0})}{(1-x)} - \frac{y^{m_2+1}(x^{m_0} - y^{m_0})}{(x-y)} \right);$$

Then we can write, under the CS policy

$$\begin{aligned} \frac{(\rho_{11} - \rho_{12})(\rho_{21} - \rho_{22})}{P(0)} &= \\ &= f_1(\rho_{11}, \rho_{21}) - f_1(\rho_{11}, \rho_{22}) - f_1(\rho_{12}, \rho_{21}) + f_1(\rho_{12}, \rho_{22}); \end{aligned}$$

and, under the SMA policy

$$\begin{aligned} \frac{(\rho_{11} - \rho_{12})(\rho_{21} - \rho_{22})}{P(0)} &= (f_2(\rho_{11}) - f_2(\rho_{12}))(f_3(\rho_{21}) - f_3(\rho_{22})) + \\ &+ f_4(\rho_{11}, \rho_{22}) - f_4(\rho_{11}, \rho_{22}) - f_4(\rho_{12}, \rho_{21}) + f_4(\rho_{12}, \rho_{22}); \end{aligned}$$

In order to state expressions for the blocking probabilities let

$$f_5(x, y) = \frac{x \cdot y (x^{M+1} - y^{M+1})}{(x-y)}$$

$$f_6(x, y) = (x^{m_1+1}) (y^{m_2+1}) \frac{(x^{m_0+1} - y^{m_0+1})}{(x-y)}$$

$$f_7(x, z) = x^{z+m_0+1}$$

$$f_8(x, y, z) = \frac{x(1-x^z)}{(1-x)}$$

We know that, under the CS policy, $B_1 = B_2 = B$ and we can state

$$B = \frac{P(Q)}{(\rho_{11}-\rho_{12})(\rho_{21}-\rho_{22})} (f_5(\rho_{11}, \rho_{12}) - f_5(\rho_{11}, \rho_{21}) - \\ - f_5(\rho_{12}, \rho_{21}) + f_5(\rho_{12}, \rho_{22}));$$

Under the SMA policy we find

$$B_1 = \frac{P(Q)}{(\rho_{11}-\rho_{12})(\rho_{21}-\rho_{22})} \cdot \\ \cdot ((f_7(\rho_{11}, m_1) - f_7(\rho_{12}, m_1))(f_8(\rho_{21}, m_2) - f_8(\rho_{22}, m_2)) + \\ + f_6(\rho_{11}, \rho_{21}) - f_6(\rho_{11}, \rho_{22}) - f_6(\rho_{12}, \rho_{21}) + f_6(\rho_{12}, \rho_{22}))$$

and

$$B_2 = \frac{P(Q)}{(\rho_{11}-\rho_{12})(\rho_{21}-\rho_{22})} \cdot \\ \cdot ((f_7(\rho_{21}, m_2) - f_7(\rho_{22}, m_2))(f_8(\rho_{11}, m_1) - f_8(\rho_{12}, m_1)) + \\ + f_6(\rho_{11}, \rho_{21}) - f_6(\rho_{11}, \rho_{22}) - f_6(\rho_{12}, \rho_{21}) + f_6(\rho_{12}, \rho_{22}));$$

8.2.3 Sample results

Let us now compare the two buffer sharing policies "Complete Sharing", (CS), and "Sharing with Minimum Allocation", (SMA), with respect to the number of blocked

(or lost) packets per time unit and the utilization of the transmission links.

a) The number of blocked packets per time unit.

In order to obtain an optimal allocation of the number of dedicated buffers m_i ($i=1,2$), i.e. the allocation that corresponds to the minimum number of lost packets, we use the linear cost function

$$f(m_1, m_2) = \sum_{i=1}^2 \lambda_i B_i$$

In order to compare the two restricted buffer sharing policies, CS and SMA, we have in figures 8.2, 8.3 and 8.4 drawn the cost-reduction

$$\frac{f(0,0) - f(m_1, m_2)}{f(0,0)}$$

as a function of ρ_{21} and with ρ_{11} as a parameter.

We furthermore assume as in all other presented results within this chapter

$$\text{i) } M = 10$$

$$\text{ii) } \frac{1}{\mu} = 10 \cdot \frac{1}{\nu}$$

$$\Rightarrow \rho_{11} = 10 \cdot \rho_{12}$$

$$\rho_{21} = 10 \cdot \rho_{22}$$

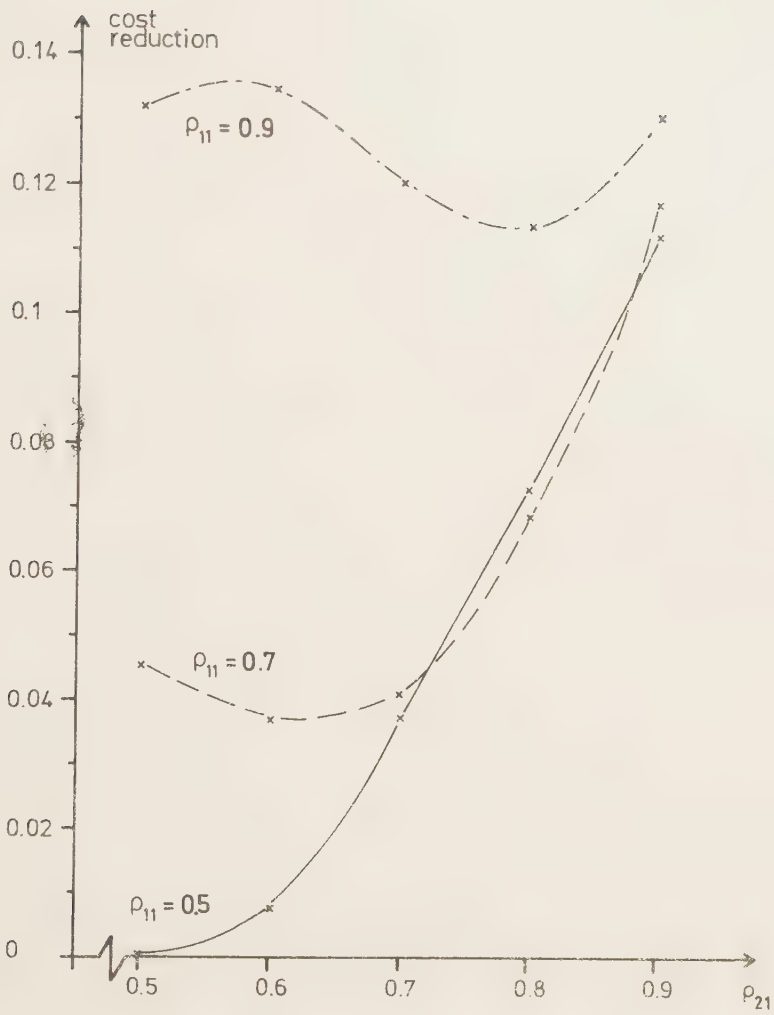


Figure 8.2

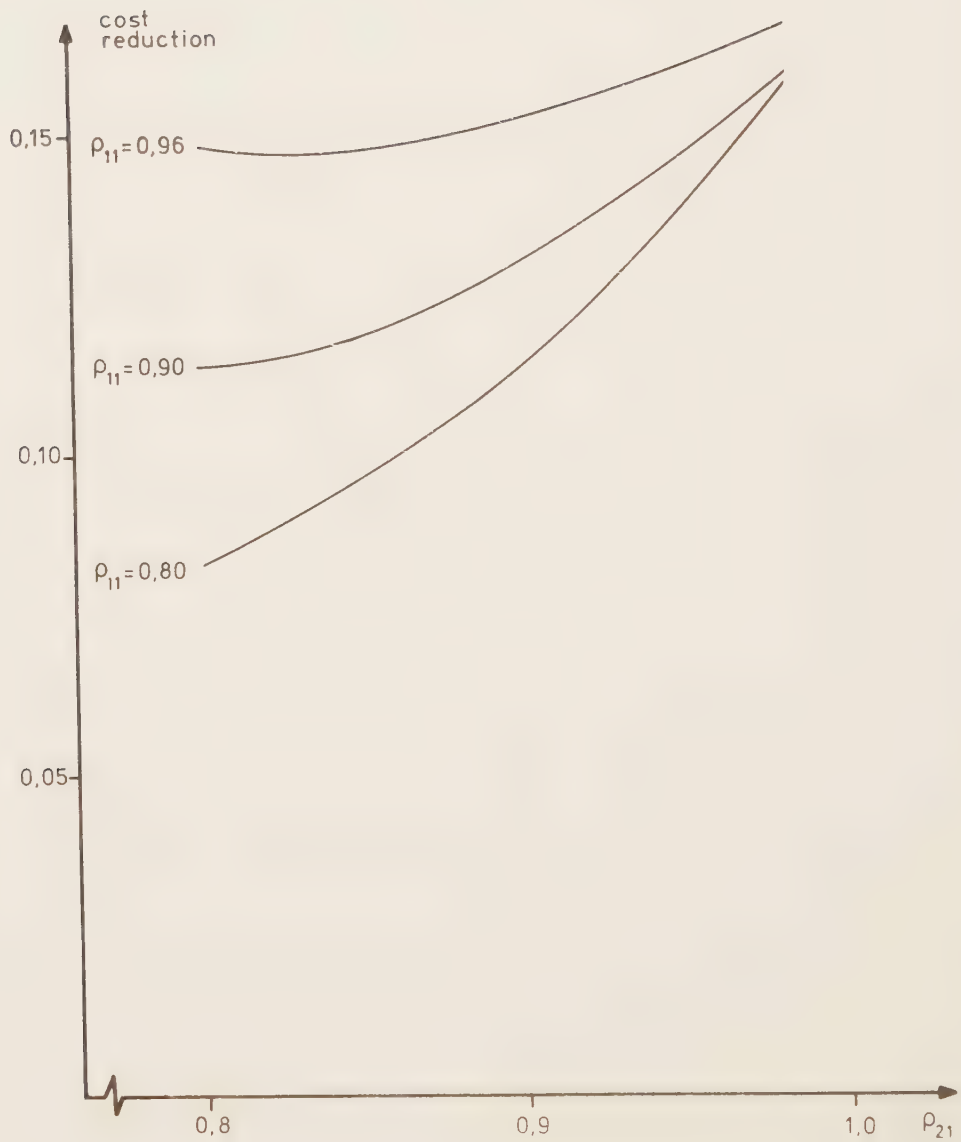


Figure 8.3

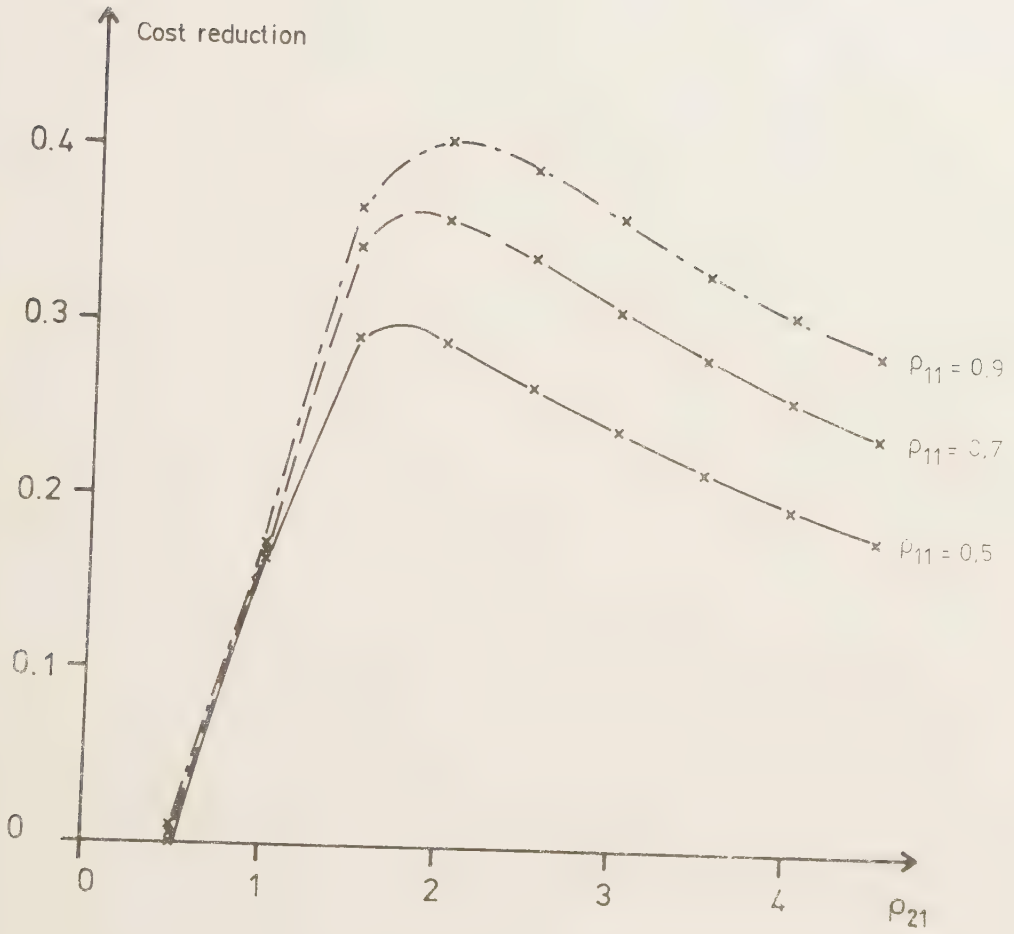


Figure 8.4

b) The number of dedicated buffers.

Figures 8.5, 8.6 and 8.7 show the number of dedicated buffers as a function of ρ_{21} with ρ_{11} as a parameter. When $C_1 = C_2$ the optimal allocation of the number of dedicated buffers can be seen as a function of the two offered loads (see chapter 6). Here we find the SMA policy allocating, as it should, more dedicated buffers to the smallest of the two offered loads.

c) Utilization or normalized throughput of the two output links.

$$U_1 = \rho_{11} (1-B_1)$$

$$U_2 = \rho_{21} (1-B_2)$$

$$U = \rho_{11} (1-B_1) + \rho_{21} (1-B_2)$$

In figure 8.8 the utilization of the output links under the CS policy is drawn. We see that total utilization or throughput decreases with increasing load. As in chapter 6.5.3, ρ_{11} is held constant and equals 0.5 and ρ_{21} is varied. The effect is, as discussed in chapter 6, that the largest traffic tends to hog all the buffers and thus prevents the smallest traffic to reach its output link.

Figure 8.9 shows the utilization of the output links under the SMA policy. Note that the utilization is derived when the optimal number of dedicated buffers is considered for every ρ_{21} . Here we find the total utilization to be a non decreasing function of offered load.

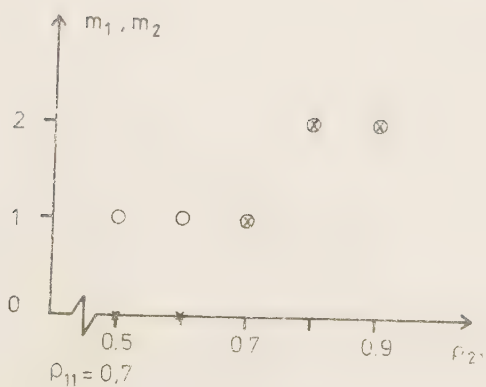
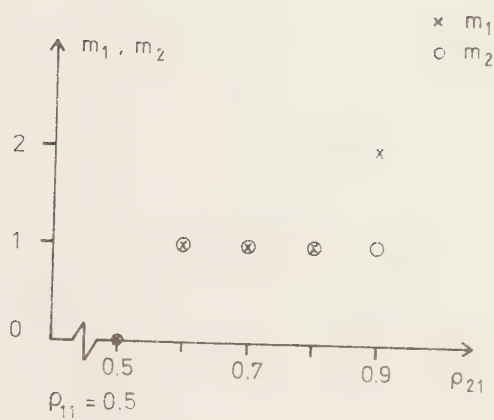


Figure 8.5

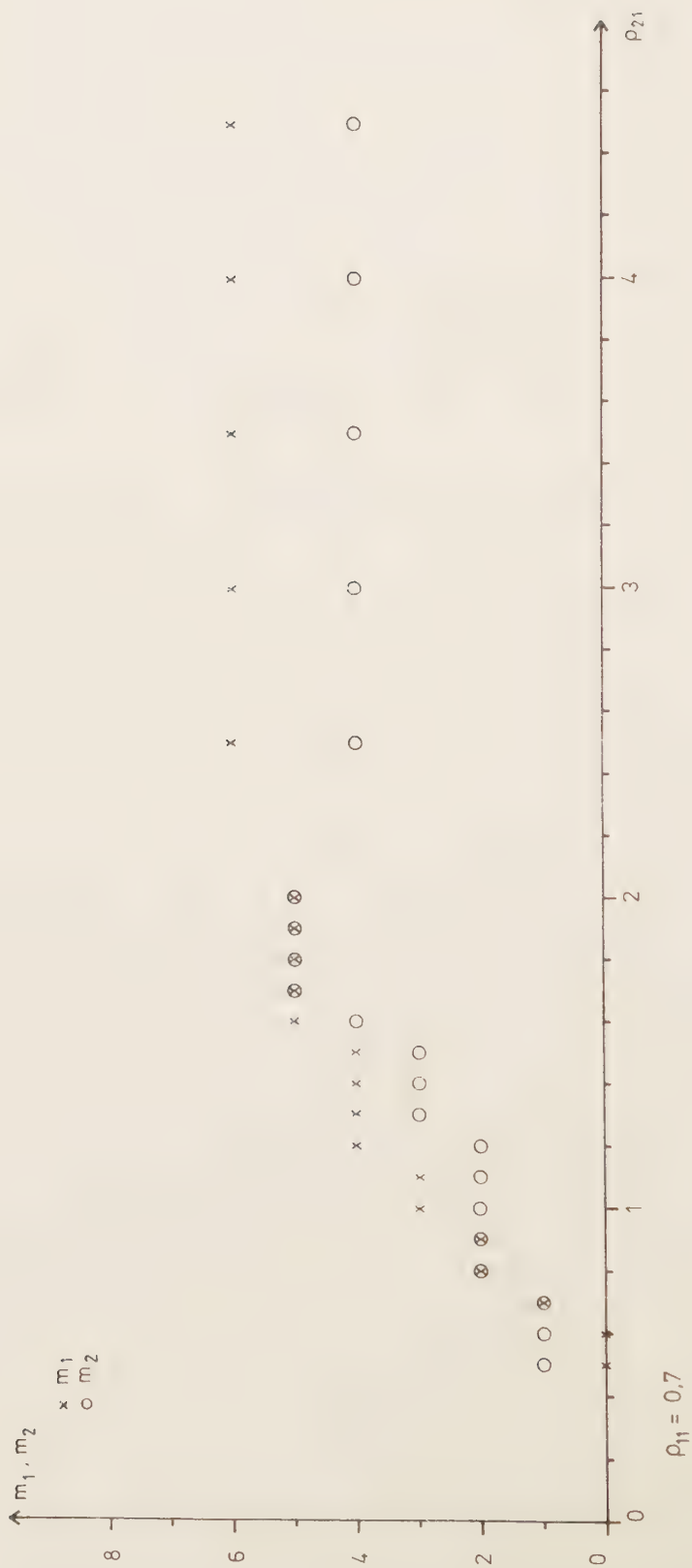


Figure 8.6

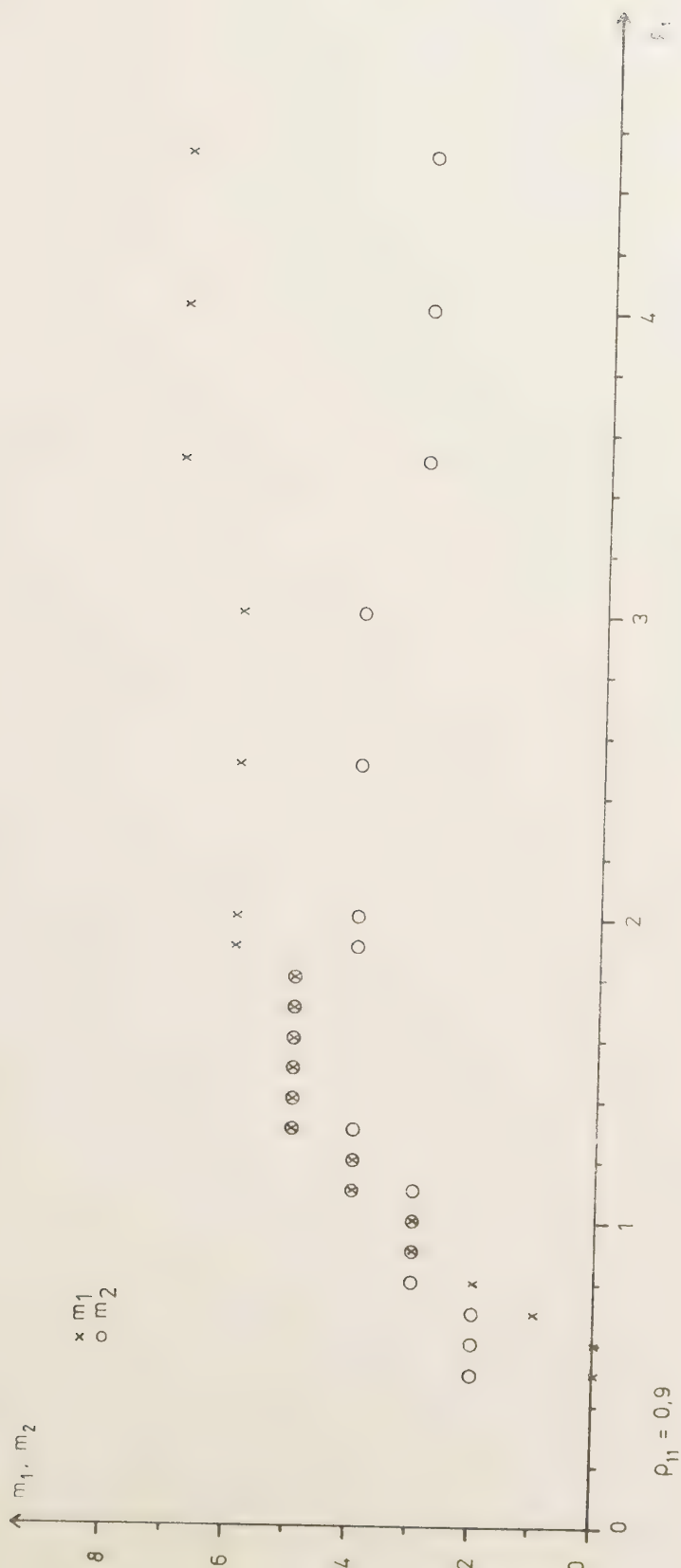


Figure 8.7

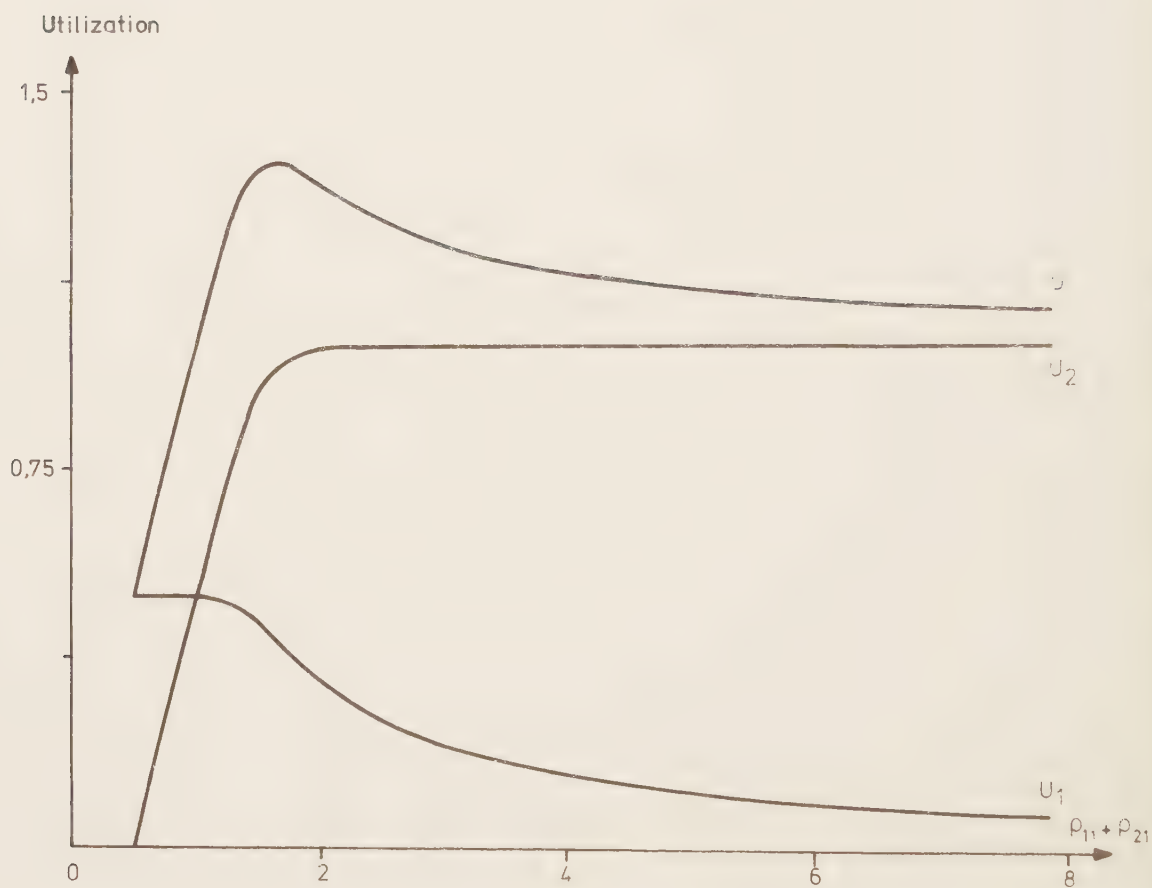


Figure 8.8 $\rho_{11} = 0.5$, CS

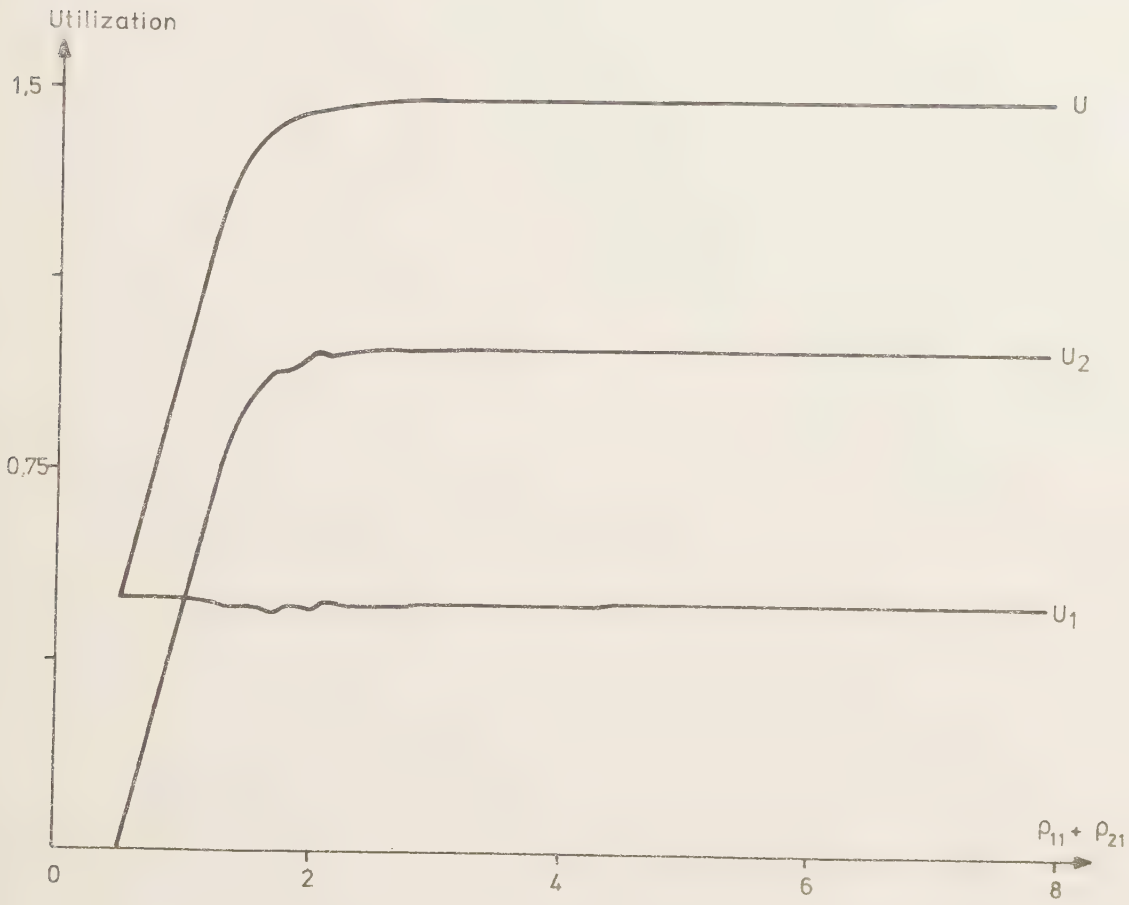


Figure 8.9 $\rho_{11} = 0.5$, SMA

8.3 A model with retransmission

Let us, with probability α , assume an acknowledgement signal to be negative and thereby to require the appropriate packet to be retransmitted. Thus a packet is held in one of the buffers during a time consisting of

- a) wait for the transmission link to become free
- b) a packet transmission time
- c) wait for an acknowledgement signal
- d) if a negative acknowledgement: a) to d) once more

8.3.1 Unlimited number of buffers

Let us for a moment consider a packet node with just one outgoing transmission link and let, as in 8.2, k denote the number of packets in the node waiting for transmission or waiting for transmission to be completed. Let furthermore ℓ denote the number of transmitted packets waiting for an acknowledgment signal and α denote the probability of an acknowledgement signal to be negative and thus requiring a retransmission. Under these assumptions we have in appendix A derived an expression, $P(k, \ell)$, for k packets waiting for transmission or for transmission to be completed and ℓ packets waiting for an acknowledgement signal. Thus the probability of having k packets in the node

$$P(k) = \sum_{\ell=0}^k P(k-\ell, \ell);$$

where

$$P(k-\ell, \ell) = \left(\frac{\lambda}{\mu C (1-\alpha)} \right)^{k-\ell} \left(\frac{\lambda}{\nu C (1-\alpha)} \right)^{\ell} P(0, 0)$$

Thus

$$P(k) = \begin{cases} \frac{\left(\frac{\lambda}{\mu C(1-\alpha)}\right)^{k+1} - \left(\frac{\lambda}{\nu C(1-\alpha)}\right)^{k+1}}{\frac{\lambda}{\mu C(1-\alpha)} - \frac{\lambda}{\nu C(1-\alpha)}} \cdot p_0 \\ (k+1) \left(\frac{\lambda}{\mu C(1-\alpha)}\right)^k \left(1 - \frac{\lambda}{\mu C(1-\alpha)}\right)^k \cdot p_0 & \mu = \nu \end{cases}$$

and

$$p_0 = \left(1 - \frac{\lambda}{\mu C(1-\alpha)}\right) \left(1 - \frac{\lambda}{\nu C(1-\alpha)}\right) \quad \mu \neq \nu$$

The observed model could be seen as two M/M/1 systems in tandem with feed-back (see figure 8.10).

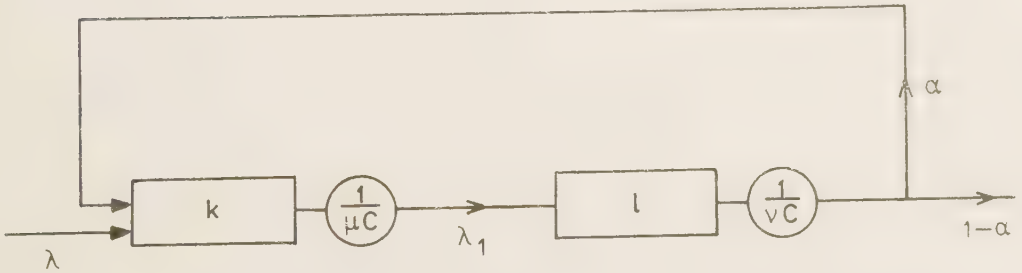


Figure 8.10

where

$$\lambda_1 = \lambda + \alpha \lambda_1 \Rightarrow \lambda_1 = \frac{\lambda}{1-\alpha}$$

$$P(k, \ell) = P(k) P(\ell)$$

$$P(k) = \rho_1^k (1 - \rho_1)$$

$$P(\ell) = \rho_2^\ell (1 - \rho_2)$$

$$\rho_1 = \frac{\lambda_1}{\mu C} \quad \rho_2 = \frac{\lambda_1}{\nu C}$$

and thus

$$P(k, \ell) = \left(\frac{\lambda}{\mu C (1-\alpha)} \right)^k \left(\frac{\lambda}{\nu C (1-\alpha)} \right)^\ell \left(1 - \frac{\lambda}{\mu C (1-\alpha)} \right) \left(1 - \frac{\lambda}{\nu C (1-\alpha)} \right);$$

8.3.2 A limited number of buffers

When we derive the steady state distribution, we use the same technique as in the unlimited case.

This method leads to

$$P(k, \ell) = \left(\frac{\lambda}{\mu C (1-\alpha)} \right)^k \left(\frac{\lambda}{\nu C (1-\alpha)} \right)^\ell P(0, 0)$$

where

$$P(0, 0) = \frac{\left(1 - \frac{\lambda}{\mu C (1-\alpha)} \right) \left(1 - \frac{\lambda}{\nu C (1-\alpha)} \right)}{\left(1 - \left(\frac{\lambda}{\mu C (1-\alpha)} \right)^{M+1} \right) \left(1 - \left(\frac{\lambda}{\nu C (1-\alpha)} \right)^{M+1} \right)};$$

We assume there are M buffers in the node.

In appendix A the solution above is chequed in the global balance equations for our two dimensional Markov process.

This model could as in 8.3.1 be seen as two M/M/1 systems in tandem, where the sum of the number of packets in the two queues may not exceed M.

We could also look upon our system as a mixed queueing network with population size constraints.

8.3.3 The general case - n transmission links

In the case where we have n outgoing transmission links, the probability of having k_i i-packets in the buffers waiting for transmission or for transmission to be completed and j_i i-packets in the buffers waiting for an acknowledgement signal ($\forall i = 1, 2, \dots, n$) is given by

$$P(\underline{k}, \underline{\ell}) = \prod_{i=1}^n \left(\frac{\lambda_i}{\mu C (1-\alpha)} \right)^{k_i} \prod_{i=1}^n \left(\frac{\lambda_i}{\nu C (1-\alpha)} \right)^{\ell_i} \cdot P(\underline{0}, \underline{0});$$

As seen in chapter 6 the principle structure of $P(\bar{k}, \bar{\ell})$ does not depend on different restricted buffer sharing policies.

Finally $P(\underline{k})$, the probability of having k_i i-packets in the buffers ($\forall i = 1, 2, \dots, n$) is given by

$$P(\underline{k}) = \sum_{\ell_1=0}^{k_1} \sum_{\ell_2=0}^{k_2} \dots \sum_{\ell_n=0}^{k_n} P(\underline{k}-\underline{\ell}, \underline{\ell});$$

8.3.4 Sample results

Figure 8.11 to 8.13 show the total as well as the individual utilization or normalized throughput of the outgoing transmission links as a function of total offered

load where ρ_{11} is held constant and equals 0.5. Figure 8.11 shows the utilization under the complete sharing policy ($m_1 = m_2 = 0$) when $\alpha = 0$ and $\alpha = 0.15$. As we saw in chapter 6, utilization of the least loaded transmission link decreases dramatically as total offered load increases. In figure 8.12 we find total utilization or throughput under the SMA policy forming a non decreasing function of offered load. Finally, in figure 8.13 we can compare utilization between the SMA and the CS policies when $\alpha = 0.15$.

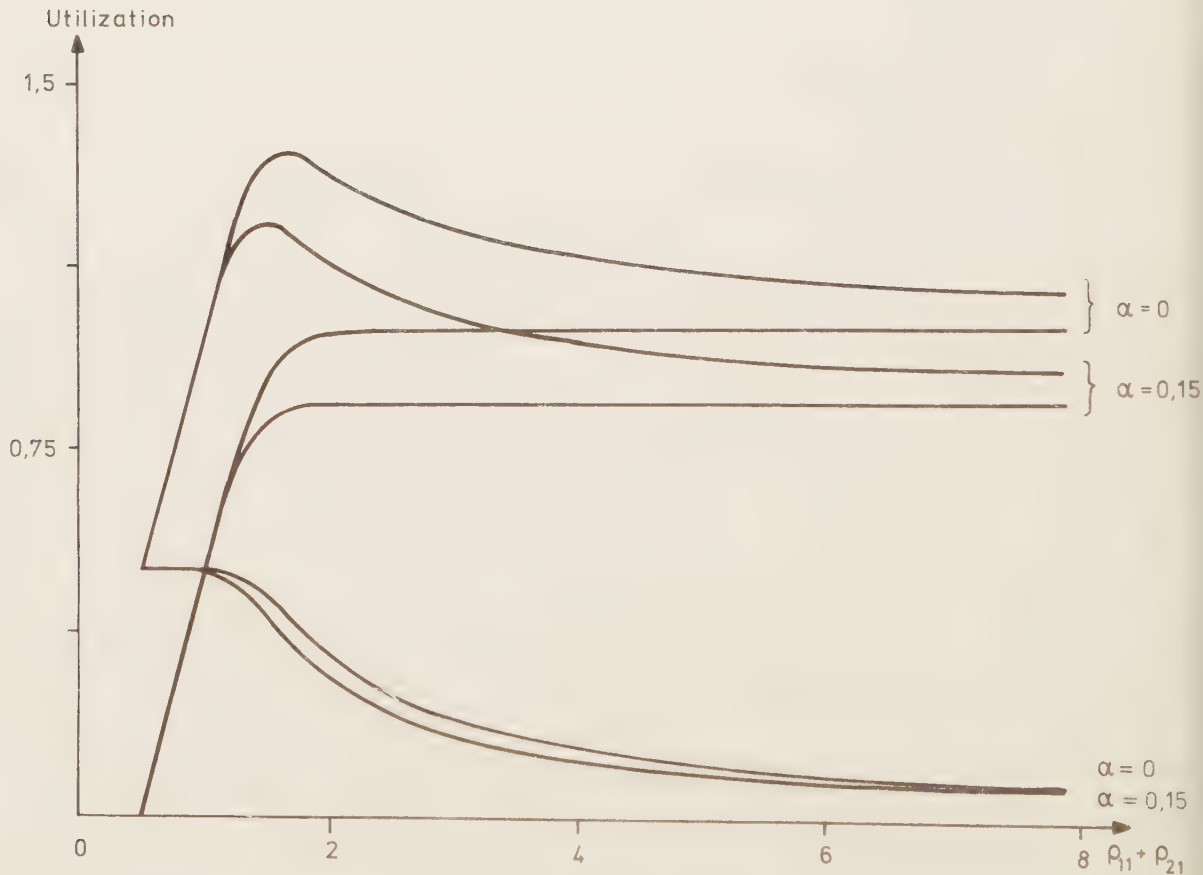


Figure 8.11 $\rho_{11} = 0.5$, CS

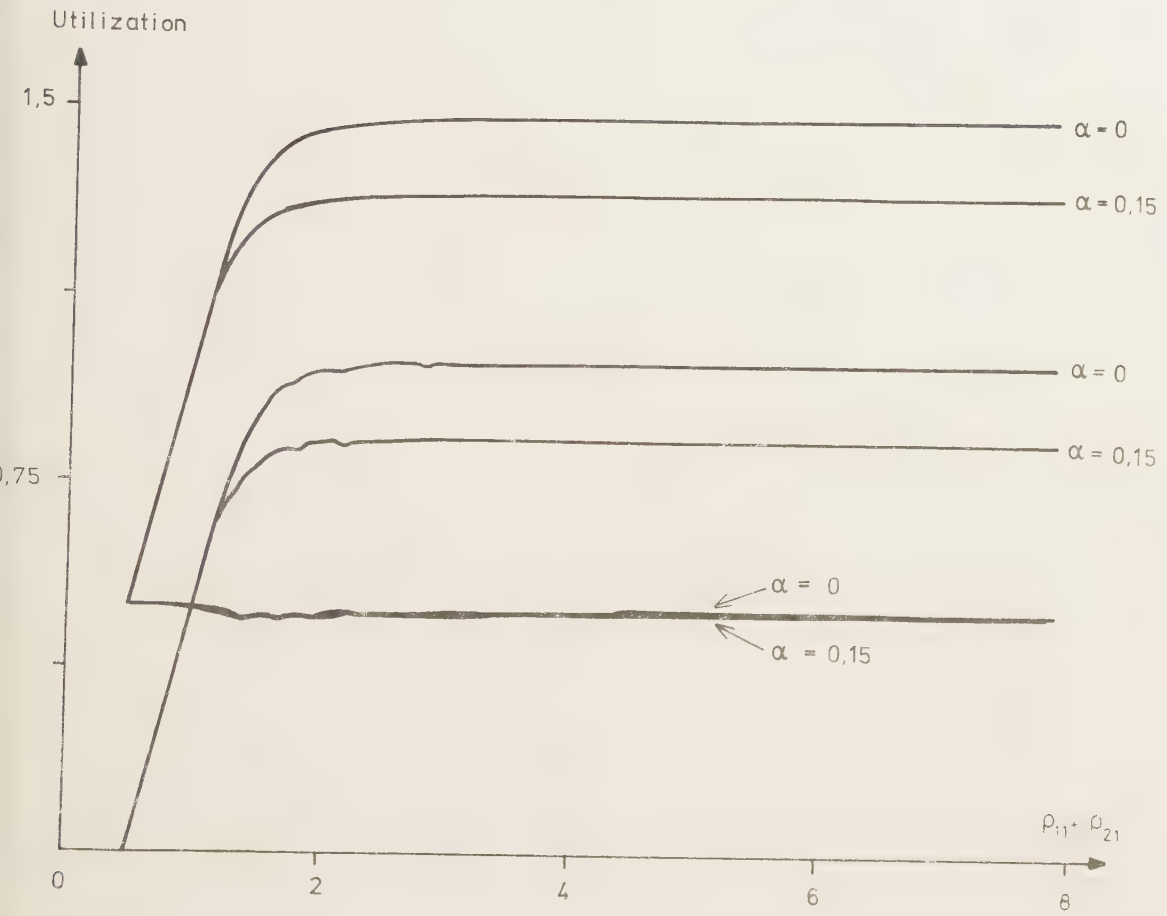


Figure 8.12 $\rho_{11} = 0.5$, SMA

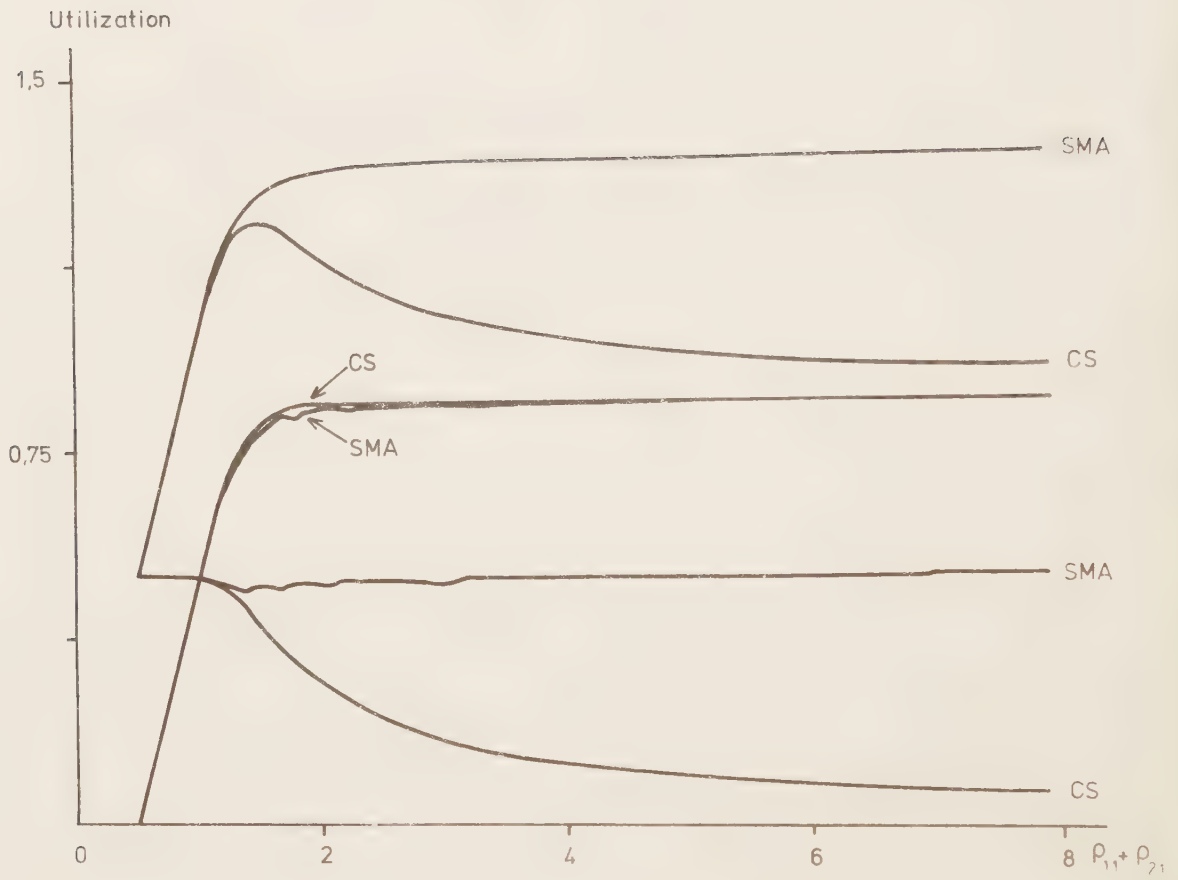


Figure 8.13 $\rho_{11} = 0.5$

8.4 Conclusions

As seen in chapter 6, the SMA policy prevents a packet stream from hogging all available buffers in a packet switching node and the utilization of the transmission links thus becomes a non decreasing function of offered load. In this chapter we have shown that we can gain the same effects when we include the effects of waiting for an acknowledgement signal and a possible retransmission.

As, in this chapter, our primary interest is to find whether the extended buffer occupancy time, in terms of waiting for an acknowledgement signal and a possible retransmission, has any principal impact on the earlier treated buffer sharing policies or not, we have only included acknowledgement signals regarding packets leaving our node. A more precise model should of course contain acknowledgement signals sent from our node as well, which contributes to the load on the outgoing transmission links. However, in this chapter we are not primarily interested in finding quantitative values, but merely a qualitative behaviour.

Schweitzer and Lam study in [85] a model of a packet switching network with finite buffer capacities and acknowledgement signals that consume part of the buffer pool. Kaul [49] studies the effects of retransmission in a model of one node with one outgoing transmission link. Butto, Colombo, Taggiasso and Tonietti [10] describe two analytic models for the evaluation of the performance of a packet switching network, which consider the effect of acknowledgement signals, retransmission and link errors. S Lam [59] derives a model of a packet node, which includes finite storage,

positive acknowledgement and time-outs. This single-node model is also included in a network model and numerical results for the performance of a 19 node network are presented.

Chapter 9 - USING PENALTY FUNCTIONS WHEN OPTIMIZING THE NUMBER OF DEDICATED BUFFERS

9.1 Introduction

In chapter 6, where we treated a model of our packet switching node, we minimized the number of lost packets per second with respect to the number of dedicated buffers. In chapter 6 we primarily used a linear cost function $f(\underline{m})$

$$f(\underline{m}) = \sum_{i=1}^n \lambda_i B_i \quad (9.1)$$

but also discussed a cost function with weight factors

$$f(\underline{m}) = \sum_{i=1}^n p_i \lambda_i B_i \quad (9.2)$$

In this chapter we will use a cost function with weight factors in order to increase total throughput in a net.

9.2 Penalty functions

When using weight factors in our cost function we prefer, in order to emphasize the use of weights, to denote it a penalty function. With penalty functions we can in general give different priorities to the different types of packets and in particular give higher priority to packets that leave the subnet at a specific node.

In our earlier models we have only included the subnet, but now we include the transmission links between our packet switch and the host computers as well.

We propose the following cost or penalty function when determining the number of dedicated buffers in node j

$$f(\underline{m}) = \sum_{i=1}^n p_i \lambda_i B_i$$

where

$$p_i = \begin{cases} \geq 1 & \text{for packets leaving the net at node } j; \\ = 1 & \text{otherwise} \end{cases}$$

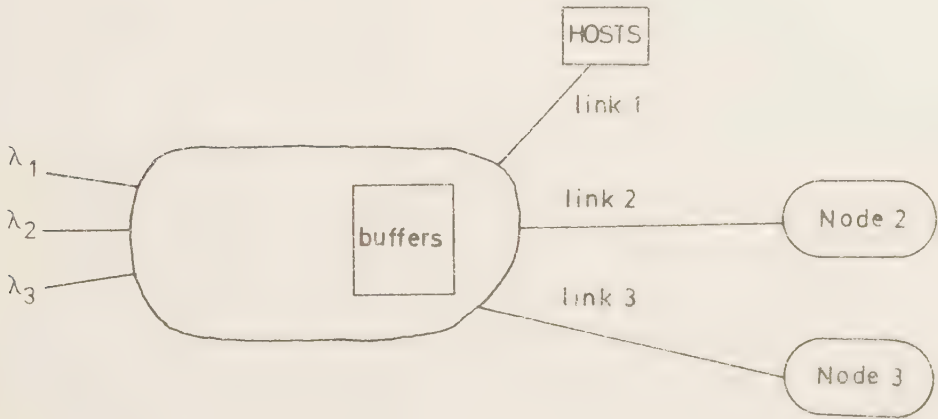


Figure 9.1

In many packet switching networks the host computer (or computers) connected to a specific node is physically placed very close to the node and is thus able to use a very fast link for communication with the node. The load on this link is consequently small compared with the load on the links that connect this particular node with other nodes. If we use a less sophisticated buffer sharing policy, say Complete Sharing (CS), we find that packets, leaving our node for other nodes, tend to hog most of the buffers and thus prevent packets bound for hosts connected to our node from reaching their hosts.

The effect gained by using weight factors in our cost function could be expressed as:

Packets in a node, bound for a host connected to that node, have, in a higher degree than other packets in the same node, used resources in the subnet in terms of transmission capacity on different links on their way from origin node to destination node (the one we deal with). Thus, using a cost function with properly chosen weights, packets which have in general gained a higher amount of resources than others, will be blocked to a minor degree.

9.3 Sample results

With the introduction of weight factors in our cost function, as discussed in the previous section, we foresee a lower blocking probability for packets bound for hosts connected to our node. This implies, however, that the blocking probabilities for packets bound for other nodes are higher, and furthermore that the total number of blocked packets in our node increases. Thus we should not choose the number of dedicated buffers that minimizes the number of lost packets per second.

In order to compare the results obtained with different weight factors and loads, we have chosen three different load patterns, each of which is divided into two cases. In the first of these two cases (case a) we assume the capacity on link 1 to be $C_1 = 1000$ kb/sec and in the second (case b) to be $C_1 = 100$ kb/sec. Thus the two cases illustrate a fast transmission link between the node and the hosts and a more "normal" respectively. The capacities on the remaining links are $C_2 = 50$ kb/sec and $C_3 = 20$ kb/sec throughout the chosen examples. Furthermore, the mean packet length is equal to 1000 bits in all three cases.

In our presented results we have held $p_2 = p_3 = 1$ and varied p_1 from 1 to 6 in steps of 1. We have also assumed the total number of buffers in our node to be 20.

Load pattern for the three cases:

I: $\lambda_1 = 21$ packets per second
 $\lambda_2 = 25$ "-
 $\lambda_3 = 11, \dots, 19$ "-

This implies that the load on link i , ρ_i is equal to

$$\rho_1 = \begin{cases} 0.021 & \text{case a} \\ 0.21 & \text{case b} \end{cases}; \rho_2 = 0.7; \rho_3 = 0.25, \dots, 0.95;$$

II: $\lambda_1 = 21$ packets per second
 $\lambda_2 = 35$ "-
 $\lambda_3 = 5, \dots, 19$ "-

In this case the load on link i , ρ_i is equal to

$$\rho_1 = \begin{cases} 0.021 & \text{case a} \\ 0.21 & \text{case b} \end{cases}; \rho_2 = 0.7; \rho_3 = 0.25, \dots, 0.95;$$

III: $\lambda_1 = 21, \dots, 45$ packets per second
 $\lambda_2 = 25, \dots, 49$ "-
 $\lambda_3 = 11, \dots, 35$ "-

The corresponding load on link i , ρ_i is equal to

$$\rho_1 = \begin{cases} 0.021, \dots, 0.045 & \text{case a} \\ 0.21, \dots, 0.45 & \text{case b} \end{cases}$$

$$\rho_2 = 0.50, \dots, 0.98$$

$$\rho_3 = 0.55, \dots, 1.75$$

Throughout the tables in this chapter column 1 and 2 show how the arrival rate(s) and load(s) are varied. Column 3, marked p_1 , shows the value on weight factor p_1 , where e g 2..6 denotes $p_1 = 2, 3, 4, 5, 6$. Column 4 ($\underline{m}$) shows the number of dedicated buffers allocated to link 1, 2 and 3, obtained by using factor p_1 in the cost function. The next three columns, marked B_1, B_2, B_3 , show the blocking probabilities on the three outgoing links and column 7, marked # lost packets, shows the total number of lost packets per second. The last two columns show the number of lost packets per second and the blocking probabilities on the links under the restriction Complete Sharing (CS).

Table 9.1 shows the results based on load pattern I, case a. We see in the first row that when p_1 in the cost function equals 1 the optimal vector $\underline{m} = \underline{0}$ and thus $B_1 = B_2 = B_3$.

When we increase p_1 the blocking probability for packets bound for hosts connected to our node is dramatically reduced. When $p_1 = 2, \dots, 6$ our strategy allocates one dedicated buffer to packets bound for link 1 and to packets bound for link 2. Thus 18 buffers are shared equally between the three links. Though link 2 has been allocated a buffer, the corresponding blocking probability is increased by almost a factor 2. This is due to the fact that the number of non dedicated buffers is decreased by two and that those buffers are used to a large extent by packets bound for link 2. Since load on link 1 is as low as 0.021, one dedicated buffer is in most cases enough and therefrom the decreased blocking probability.

Compare with a simple M/M/1.20 buffers system in which

the probability of finding more than one buffer occupied is for $\rho = 0.021$ $4.411 \cdot 10^{-4}$, and for $\rho = 0.5$ $2.50 \cdot 10^{-1}$.

We can also see that except for $\lambda_3 = 11$, the SMA policy favours smaller loads.

The increase of the total number of lost packets, when we increase p_1 , is, if we compare with Complete Sharing ($\underline{m} = 0$), tolerable.

λ_3	ρ_3	p_1	$\underline{m}$	B_1	B_2	B_3	# lost packets	$\underline{m} = 0$ # lost packets	$\underline{m} = \underline{1}$ $B_1 + B_2 + B_3$
11	0.55	1	0,0,0	$1.40 \cdot 10^{-5}$	$1.40 \cdot 10^{-5}$	$1.40 \cdot 10^{-5}$	$5.96 \cdot 10^{-4}$	$7.96 \cdot 10^{-4}$	$1.43 \cdot 10^{-1}$
		2..6	1,1,0	$5.71 \cdot 10^{-7}$	$2.20 \cdot 10^{-5}$	$2.67 \cdot 10^{-5}$	$8.55 \cdot 10^{-4}$		
12	0.60	1	1,2,0	$2.12 \cdot 10^{-6}$	$4.89 \cdot 10^{-5}$	$9.97 \cdot 10^{-5}$	$2.46 \cdot 10^{-3}$	$2.52 \cdot 10^{-3}$	$4.35 \cdot 10^{-5}$
		2..6	1,1,0	$1.69 \cdot 10^{-6}$	$5.91 \cdot 10^{-5}$	$7.94 \cdot 10^{-5}$	$2.46 \cdot 10^{-3}$		
13	0.65	1..5	1,2,0	$6.32 \cdot 10^{-6}$	$1.24 \cdot 10^{-4}$	$2.97 \cdot 10^{-4}$	$7.10 \cdot 10^{-3}$	$8.17 \cdot 10^{-3}$	$1.39 \cdot 10^{-4}$
		6	1,1,0	$5.03 \cdot 10^{-6}$	$1.62 \cdot 10^{-4}$	$2.37 \cdot 10^{-4}$	$7.23 \cdot 10^{-3}$		
14	0.70	1..6	1,2,0	$1.76 \cdot 10^{-5}$	$3.05 \cdot 10^{-4}$	$8.30 \cdot 10^{-4}$	$1.96 \cdot 10^{-2}$	$2.33 \cdot 10^{-2}$	$4.23 \cdot 10^{-4}$
15	0.75	1..6	1,2,0	$4.50 \cdot 10^{-5}$	$7.09 \cdot 10^{-4}$	$2.13 \cdot 10^{-3}$	$5.08 \cdot 10^{-2}$	$7.33 \cdot 10^{-2}$	$1.20 \cdot 10^{-3}$
16	0.80	1..6	1,2,0	$1.05 \cdot 10^{-4}$	$1.53 \cdot 10^{-3}$	$4.98 \cdot 10^{-3}$	$1.20 \cdot 10^{-1}$	$1.94 \cdot 10^{-1}$	$3.13 \cdot 10^{-3}$
17	0.85	1..6	1,3,0	$2.58 \cdot 10^{-4}$	$1.81 \cdot 10^{-3}$	$1.22 \cdot 10^{-2}$	$2.48 \cdot 10^{-1}$	$4.65 \cdot 10^{-1}$	$1.38 \cdot 10^{-3}$
18	0.90	1..6	1,3,0	$4.77 \cdot 10^{-4}$	$3.14 \cdot 10^{-3}$	$2.27 \cdot 10^{-2}$	$4.96 \cdot 10^{-1}$	$1.00 \cdot 10^0$	$1.56 \cdot 10^{-2}$
19	0.95	1..6	1,3,0	$8.05 \cdot 10^{-4}$	$5.02 \cdot 10^{-3}$	$3.83 \cdot 10^{-2}$	$8.96 \cdot 10^{-1}$	$1.92 \cdot 10^0$	$2.96 \cdot 10^{-2}$

Table 9.1

Table 9.2, based on load pattern I, case b, shows a result different from the former. In this case the load on link 1 is equal to 0.21. A comparison with our earlier M/M/1.20 buffer system yields the probability of finding more than 1 packet in the system to be equal to $4.41 \cdot 10^{-2}$ for $\rho_1 = 0.21$. The result of this effect could be seen in table 9.2 if we compare the two blocking probabilities for packets bound for link 1 when $p_1 = 1$ and $p_2 = 2..3$ respectively. The dramatically decrease of

the blocking probability found in table 9.1 does not appear here.

Furthermore we find, when comparing table 9.1 and 9.2, that the SMA policy does not, in the second table, favour packets bound for link 1 as the load on link 1 in table 9.2 is higher than the corresponding load in table 9.1.

λ_3	ρ_3	P_1	$\underline{m}$	B_1	B_2	B_3	# lost packets	$\underline{m} = 0$ # lost packets	$\underline{m} = 0$ $B_1 = B_2 = B_3 = 0$
11	0.55	1	0,0,0	$1.74 \cdot 10^{-5}$	$1.74 \cdot 10^{-5}$	$1.74 \cdot 10^{-5}$	$9.90 \cdot 10^{-4}$	$9.90 \cdot 10^{-4}$	$1.74 \cdot 10^{-5}$
		2..3	1,1,0	$7.10 \cdot 10^{-6}$	$2.32 \cdot 10^{-5}$	$2.82 \cdot 10^{-5}$	$1.04 \cdot 10^{-3}$		
		4..6	1,0,0	$6.52 \cdot 10^{-6}$	$2.59 \cdot 10^{-5}$	$2.59 \cdot 10^{-5}$	$1.07 \cdot 10^{-3}$		
12	0.60	1..4	1,1,0	$2.02 \cdot 10^{-5}$	$6.16 \cdot 10^{-5}$	$8.29 \cdot 10^{-5}$	$2.96 \cdot 10^{-3}$	$3.02 \cdot 10^{-3}$	$5.20 \cdot 10^{-5}$
		5	1,0,0	$1.81 \cdot 10^{-5}$	$7.44 \cdot 10^{-5}$	$7.44 \cdot 10^{-5}$	$3.13 \cdot 10^{-3}$		
		6	2,1,0	$7.03 \cdot 10^{-6}$	$9.87 \cdot 10^{-5}$	$1.33 \cdot 10^{-4}$	$4.21 \cdot 10^{-3}$		
13	0.65	1..4	1,1,0	$5.80 \cdot 10^{-5}$	$1.67 \cdot 10^{-4}$	$2.45 \cdot 10^{-4}$	$8.58 \cdot 10^{-3}$	$9.42 \cdot 10^{-3}$	$1.60 \cdot 10^{-4}$
		5..6	2,1,0	$1.87 \cdot 10^{-5}$	$2.50 \cdot 10^{-4}$	$3.66 \cdot 10^{-4}$	$1.14 \cdot 10^{-2}$		
14	0.70	1	1,2,0	$1.97 \cdot 10^{-4}$	$3.13 \cdot 10^{-4}$	$8.52 \cdot 10^{-4}$	$2.39 \cdot 10^{-2}$	$2.84 \cdot 10^{-2}$	$4.73 \cdot 10^{-4}$
		2..3	1,1,0	$1.59 \cdot 10^{-4}$	$4.39 \cdot 10^{-4}$	$6.90 \cdot 10^{-4}$	$2.40 \cdot 10^{-2}$		
		4..5	2,2,0	$5.90 \cdot 10^{-4}$	$4.38 \cdot 10^{-4}$	$1.19 \cdot 10^{-3}$	$2.89 \cdot 10^{-2}$		
		6	2,1,0	$4.78 \cdot 10^{-5}$	$6.14 \cdot 10^{-4}$	$9.66 \cdot 10^{-4}$	$2.99 \cdot 10^{-2}$		
15	0.75	1..2	1,2,0	$4.90 \cdot 10^{-4}$	$7.23 \cdot 10^{-4}$	$2.17 \cdot 10^{-3}$	$6.09 \cdot 10^{-2}$	$7.99 \cdot 10^{-2}$	$1.31 \cdot 10^{-3}$
		3..6	2,2,0	$1.38 \cdot 10^{-4}$	$9.51 \cdot 10^{-4}$	$2.85 \cdot 10^{-3}$	$6.95 \cdot 10^{-2}$		
16	0.80	1	1,2,0	$1.12 \cdot 10^{-3}$	$1.55 \cdot 10^{-3}$	$5.05 \cdot 10^{-3}$	$1.43 \cdot 10^{-1}$	$2.07 \cdot 10^{-1}$	$3.34 \cdot 10^{-3}$
		2..6	2,2,0	$2.95 \cdot 10^{-4}$	$1.93 \cdot 10^{-3}$	$6.27 \cdot 10^{-3}$	$1.55 \cdot 10^{-1}$		
17	0.85	1	1,2,0	$2.33 \cdot 10^{-3}$	$3.08 \cdot 10^{-3}$	$1.07 \cdot 10^{-2}$	$3.08 \cdot 10^{-1}$	$4.87 \cdot 10^{-1}$	$7.73 \cdot 10^{-3}$
		2..6	2,2,0	$5.82 \cdot 10^{-4}$	$3.63 \cdot 10^{-3}$	$1.26 \cdot 10^{-2}$	$3.17 \cdot 10^{-1}$		
18	0.90	1..4	2,3,0	$1.17 \cdot 10^{-3}$	$3.57 \cdot 10^{-3}$	$2.58 \cdot 10^{-2}$	$5.78 \cdot 10^{-1}$	$1.03 \cdot 10^0$	$1.61 \cdot 10^{-2}$
		5..6	3,3,0	$2.80 \cdot 10^{-4}$	$4.07 \cdot 10^{-3}$	$2.94 \cdot 10^{-2}$	$6.37 \cdot 10^{-1}$		
19	0.95	1..3	2,3,0	$1.87 \cdot 10^{-3}$	$5.50 \cdot 10^{-3}$	$4.19 \cdot 10^{-2}$	$9.74 \cdot 10^{-1}$	$1.96 \cdot 10^0$	$3.02 \cdot 10^{-2}$
		4..6	3,3,0	$4.32 \cdot 10^{-4}$	$6.03 \cdot 10^{-3}$	$4.60 \cdot 10^{-2}$	$1.03 \cdot 10^0$		

Table 9.2

In tables 9.3 and 9.4, based on load pattern II, case a and b respectively, we find the SMA policy with $p_1 = 1$ to be a rather good policy when the load on link 1 is small (case a, table 9.3). In most cases the optimal allocation of dedicated buffers does not differ when $p_1 = 1$ and $p_1 > 1$. In table 9.4, which corresponds to a load on link 1 of 0.21, we find that the decrease of blocking probability B_1 (when p_1 increase) is not that large. For higher values of ρ_3 the decrease is a little bit more significant.

λ_3	ρ_3	p_1	m	P_1	B_2	B_1	# lost packets	$\frac{m - c}{\# \text{ lost packets}}$	$\frac{c}{B_1=B_2=B_3}$
5	0.25	1..6	1,1,0	$8.46 \cdot 10^{-6}$	$3.99 \cdot 10^{-4}$	$3.99 \cdot 10^{-1}$	$1.62 \cdot 10^{-2}$	$1.72 \cdot 10^{-2}$	$2.82 \cdot 10^{-4}$
6	0.30	1..6	1,1,0	$8.89 \cdot 10^{-6}$	$4.19 \cdot 10^{-4}$	$4.19 \cdot 10^{-4}$	$1.74 \cdot 10^{-2}$	$1.84 \cdot 10^{-2}$	$2.96 \cdot 10^{-4}$
7	0.35	1	1,2,0	$9.43 \cdot 10^{-6}$	$4.45 \cdot 10^{-4}$	$4.45 \cdot 10^{-4}$	$1.89 \cdot 10^{-2}$	$1.98 \cdot 10^{-2}$	$3.04 \cdot 10^{-4}$
		2..6	1,1,0	$9.43 \cdot 10^{-6}$	$4.45 \cdot 10^{-4}$	$4.45 \cdot 10^{-4}$	$1.89 \cdot 10^{-2}$		
8	0.40	1..3	1,2,0	$1.02 \cdot 10^{-5}$	$4.79 \cdot 10^{-4}$	$4.74 \cdot 10^{-4}$	$2.08 \cdot 10^{-2}$	$2.17 \cdot 10^{-2}$	$3.39 \cdot 10^{-4}$
		4..6	1,1,0	$1.02 \cdot 10^{-5}$	$4.79 \cdot 10^{-4}$	$4.79 \cdot 10^{-4}$	$2.08 \cdot 10^{-2}$		
9	0.45	1..6	1,2,0	$1.12 \cdot 10^{-5}$	$5.27 \cdot 10^{-4}$	$5.28 \cdot 10^{-4}$	$2.34 \cdot 10^{-2}$	$2.42 \cdot 10^{-2}$	$3.72 \cdot 10^{-4}$
10	0.50	1..6	1,2,0	$1.27 \cdot 10^{-5}$	$5.98 \cdot 10^{-4}$	$6.00 \cdot 10^{-4}$	$2.72 \cdot 10^{-2}$	$2.79 \cdot 10^{-2}$	$4.23 \cdot 10^{-4}$
11	0.55	1..6	1,2,0	$1.52 \cdot 10^{-5}$	$7.10 \cdot 10^{-4}$	$7.19 \cdot 10^{-4}$	$3.31 \cdot 10^{-2}$	$3.38 \cdot 10^{-2}$	$5.05 \cdot 10^{-4}$
12	0.60	1..6	1,2,0	$1.98 \cdot 10^{-5}$	$9.06 \cdot 10^{-4}$	$9.18 \cdot 10^{-4}$	$4.31 \cdot 10^{-2}$	$4.43 \cdot 10^{-2}$	$6.11 \cdot 10^{-4}$
13	0.65	1..3	1,3,0	$3.05 \cdot 10^{-5}$	$1.20 \cdot 10^{-3}$	$1.44 \cdot 10^{-3}$	$6.15 \cdot 10^{-2}$	$6.46 \cdot 10^{-2}$	$9.36 \cdot 10^{-4}$
		4..6	1,2,0	$2.87 \cdot 10^{-5}$	$1.24 \cdot 10^{-3}$	$1.36 \cdot 10^{-3}$	$6.16 \cdot 10^{-2}$		
15	0.75	1..6	1,3,0	$9.25 \cdot 10^{-5}$	$2.71 \cdot 10^{-3}$	$4.38 \cdot 10^{-3}$	$1.62 \cdot 10^{-1}$	$1.97 \cdot 10^{-1}$	$2.77 \cdot 10^{-3}$
16	0.80	1..3	1,4,1	$1.96 \cdot 10^{-4}$	$3.74 \cdot 10^{-3}$	$9.19 \cdot 10^{-3}$	$2.82 \cdot 10^{-1}$	$3.84 \cdot 10^{-1}$	$5.34 \cdot 10^{-3}$
		4	1,4,0	$1.96 \cdot 10^{-4}$	$3.71 \cdot 10^{-3}$	$9.26 \cdot 10^{-3}$	$2.82 \cdot 10^{-1}$		
		5..6	1,3,0	$1.71 \cdot 10^{-4}$	$4.29 \cdot 10^{-3}$	$8.12 \cdot 10^{-3}$	$2.84 \cdot 10^{-1}$		
17	0.85	1..6	1,4,1	$3.48 \cdot 10^{-4}$	$5.62 \cdot 10^{-3}$	$1.64 \cdot 10^{-2}$	$4.83 \cdot 10^{-1}$	$7.54 \cdot 10^{-1}$	$1.03 \cdot 10^{-2}$
18	0.90	1..3	1,5,1	$6.49 \cdot 10^{-4}$	$6.50 \cdot 10^{-3}$	$3.08 \cdot 10^{-2}$	$7.95 \cdot 10^{-1}$	$1.42 \cdot 10^0$	$1.92 \cdot 10^{-2}$
		4..6	1,4,1	$5.85 \cdot 10^{-4}$	$8.22 \cdot 10^{-3}$	$2.77 \cdot 10^{-2}$	$7.99 \cdot 10^{-1}$		
19	0.95	1..6	1,5,1	$9.96 \cdot 10^{-4}$	$8.81 \cdot 10^{-3}$	$4.73 \cdot 10^{-2}$	$1.23 \cdot 10^0$	$2.51 \cdot 10^0$	$3.34 \cdot 10^{-2}$

Table 9.3

λ_3	ϵ_3	P_1	$\underline{m}$	B_1	B_2	B_3	# lost packets	$\frac{m}{n} = 0$ # lost packets	$\frac{m}{n} = 0$ $B_1 = B_2 = B_3$
5	0.25	1	1,1,0	$9.46 \cdot 10^{-5}$	$4.10 \cdot 10^{-4}$	$4.10 \cdot 10^{-4}$	$1.84 \cdot 10^{-2}$	$1.92 \cdot 10^{-2}$	$3.15 \cdot 10^{-4}$
		2..4	1,0,0	$9.46 \cdot 10^{-5}$	$4.10 \cdot 10^{-4}$	$4.10 \cdot 10^{-4}$	$1.84 \cdot 10^{-2}$		
		5..6	2,0,0	$2.84 \cdot 10^{-5}$	$5.74 \cdot 10^{-4}$	$5.74 \cdot 10^{-4}$	$2.36 \cdot 10^{-2}$		
6	0.30	1	1,1,0	$9.94 \cdot 10^{-5}$	$4.31 \cdot 10^{-4}$	$4.31 \cdot 10^{-4}$	$1.97 \cdot 10^{-2}$	$2.05 \cdot 10^{-2}$	$3.31 \cdot 10^{-4}$
		2..4	1,0,0	$9.94 \cdot 10^{-5}$	$4.31 \cdot 10^{-4}$	$4.31 \cdot 10^{-4}$	$1.97 \cdot 10^{-2}$		
		5..6	2,0,0	$2.98 \cdot 10^{-5}$	$6.03 \cdot 10^{-4}$	$6.03 \cdot 10^{-4}$	$2.53 \cdot 10^{-2}$		
7	0.35	1	1,1,0	$1.05 \cdot 10^{-4}$	$4.57 \cdot 10^{-4}$	$4.57 \cdot 10^{-4}$	$2.14 \cdot 10^{-2}$	$2.21 \cdot 10^{-2}$	$3.51 \cdot 10^{-4}$
		2..4	1,0,0	$1.05 \cdot 10^{-4}$	$4.57 \cdot 10^{-4}$	$4.57 \cdot 10^{-4}$	$2.14 \cdot 10^{-2}$		
		5..6	2,1,0	$3.17 \cdot 10^{-5}$	$6.40 \cdot 10^{-4}$	$6.40 \cdot 10^{-4}$	$2.75 \cdot 10^{-2}$		
8	0.40	1..2	1,1,0	$1.14 \cdot 10^{-4}$	$4.92 \cdot 10^{-4}$	$4.92 \cdot 10^{-4}$	$2.36 \cdot 10^{-2}$	$2.42 \cdot 10^{-2}$	$3.79 \cdot 10^{-4}$
		3..5	1,0,0	$1.14 \cdot 10^{-4}$	$4.92 \cdot 10^{-4}$	$4.92 \cdot 10^{-4}$	$2.36 \cdot 10^{-2}$		
		6	2,1,0	$3.41 \cdot 10^{-5}$	$6.89 \cdot 10^{-4}$	$6.89 \cdot 10^{-4}$	$3.03 \cdot 10^{-2}$		
9	0.45	1..3	1,1,0	$1.25 \cdot 10^{-4}$	$5.41 \cdot 10^{-4}$	$5.42 \cdot 10^{-4}$	$2.64 \cdot 10^{-2}$	$2.71 \cdot 10^{-2}$	$4.16 \cdot 10^{-4}$
		4..5	1,0,0	$1.25 \cdot 10^{-4}$	$5.41 \cdot 10^{-4}$	$5.41 \cdot 10^{-4}$	$2.64 \cdot 10^{-2}$		
		6	2,1,0	$3.75 \cdot 10^{-5}$	$7.58 \cdot 10^{-4}$	$7.58 \cdot 10^{-4}$	$3.41 \cdot 10^{-2}$		
10	0.50	1..4	1,1,0	$1.42 \cdot 10^{-4}$	$6.14 \cdot 10^{-4}$	$6.15 \cdot 10^{-4}$	$3.06 \cdot 10^{-2}$	$3.12 \cdot 10^{-2}$	$4.73 \cdot 10^{-4}$
		5	1,0,0	$1.42 \cdot 10^{-4}$	$6.15 \cdot 10^{-4}$	$6.15 \cdot 10^{-4}$	$3.06 \cdot 10^{-2}$		
		6	2,1,0	$4.26 \cdot 10^{-5}$	$8.59 \cdot 10^{-4}$	$8.61 \cdot 10^{-4}$	$3.96 \cdot 10^{-2}$		
11	0.55	1	1,2,0	$1.70 \cdot 10^{-4}$	$7.29 \cdot 10^{-4}$	$7.38 \cdot 10^{-4}$	$3.72 \cdot 10^{-2}$	$3.78 \cdot 10^{-2}$	$5.64 \cdot 10^{-4}$
		2..5	1,1,0	$1.69 \cdot 10^{-4}$	$7.31 \cdot 10^{-4}$	$7.34 \cdot 10^{-4}$	$7.72 \cdot 10^{-2}$		
		6	2,1,0	$5.07 \cdot 10^{-5}$	$1.02 \cdot 10^{-3}$	$1.03 \cdot 10^{-3}$	$4.81 \cdot 10^{-2}$		
12	0.60	1	1,2,0	$2.21 \cdot 10^{-4}$	$9.23 \cdot 10^{-4}$	$9.59 \cdot 10^{-4}$	$4.85 \cdot 10^{-2}$	$4.93 \cdot 10^{-2}$	$7.26 \cdot 10^{-4}$
		2..5	1,1,0	$2.17 \cdot 10^{-4}$	$9.32 \cdot 10^{-4}$	$9.45 \cdot 10^{-4}$	$4.65 \cdot 10^{-2}$		
		6	2,2,0	$6.60 \cdot 10^{-5}$	$1.28 \cdot 10^{-3}$	$1.34 \cdot 10^{-3}$	$6.22 \cdot 10^{-2}$		
13	0.65	1..2	1,2,0	$3.19 \cdot 10^{-4}$	$1.27 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	$6.92 \cdot 10^{-2}$	$7.16 \cdot 10^{-2}$	$1.04 \cdot 10^{-3}$
		3..4	1,1,0	$3.09 \cdot 10^{-4}$	$1.30 \cdot 10^{-3}$	$1.35 \cdot 10^{-3}$	$6.95 \cdot 10^{-2}$		
		5..6	2,2,0	$9.40 \cdot 10^{-5}$	$1.73 \cdot 10^{-3}$	$1.91 \cdot 10^{-3}$	$8.75 \cdot 10^{-2}$		
15	0.75	1	1,3,0	$1.00 \cdot 10^{-3}$	$2.76 \cdot 10^{-3}$	$4.46 \cdot 10^{-3}$	$1.84 \cdot 10^{-1}$	$2.13 \cdot 10^{-1}$	$3.33 \cdot 10^{-3}$
		2..3	1,2,0	$9.03 \cdot 10^{-4}$	$3.04 \cdot 10^{-3}$	$4.03 \cdot 10^{-3}$	$1.86 \cdot 10^{-1}$		
		4..6	2,3,0	$2.76 \cdot 10^{-4}$	$3.52 \cdot 10^{-3}$	$5.77 \cdot 10^{-3}$	$2.16 \cdot 10^{-1}$		
16	0.80	1..2	1,3,0	$1.82 \cdot 10^{-3}$	$4.35 \cdot 10^{-3}$	$8.23 \cdot 10^{-3}$	$3.22 \cdot 10^{-1}$	$4.09 \cdot 10^{-1}$	$5.88 \cdot 10^{-3}$
		3..6	2,3,0	$4.77 \cdot 10^{-4}$	$5.34 \cdot 10^{-3}$	$1.02 \cdot 10^{-2}$	$3.60 \cdot 10^{-1}$		
17	0.85	1	1,4,0	$3.63 \cdot 10^{-3}$	$8.65 \cdot 10^{-3}$	$1.67 \cdot 10^{-2}$	$8.57 \cdot 10^{-1}$	$7.89 \cdot 10^{-1}$	$1.28 \cdot 10^{-2}$
		2..3	2,4,1	$9.10 \cdot 10^{-4}$	$6.62 \cdot 10^{-3}$	$1.96 \cdot 10^{-2}$	$8.86 \cdot 10^{-1}$		
		4..6	2,4,0	$9.08 \cdot 10^{-4}$	$6.64 \cdot 10^{-3}$	$1.97 \cdot 10^{-2}$	$5.26 \cdot 10^{-1}$		
18	0.90	1	1,4,1	$6.02 \cdot 10^{-3}$	$8.28 \cdot 10^{-3}$	$2.79 \cdot 10^{-2}$	$9.19 \cdot 10^{-1}$	$1.47 \cdot 10^0$	$1.96 \cdot 10^{-2}$
		2..4	2,4,1	$1.44 \cdot 10^{-3}$	$9.36 \cdot 10^{-3}$	$3.17 \cdot 10^{-2}$	$9.28 \cdot 10^{-1}$		
		5	2,4,0	$1.44 \cdot 10^{-3}$	$9.34 \cdot 10^{-3}$	$3.17 \cdot 10^{-2}$	$9.28 \cdot 10^{-1}$		
19	0.95	1	3,4,1	$3.46 \cdot 10^{-4}$	$1.07 \cdot 10^{-2}$	$3.62 \cdot 10^{-2}$	$1.03 \cdot 10^0$	$2.36 \cdot 10^0$	$3.41 \cdot 10^{-3}$
		2..3	2,4,1	$2.34 \cdot 10^{-3}$	$9.71 \cdot 10^{-3}$	$5.23 \cdot 10^{-2}$	$1.38 \cdot 10^0$		
		4..6	3,4,1	$5.45 \cdot 10^{-4}$	$1.07 \cdot 10^{-2}$	$5.80 \cdot 10^{-2}$	$1.49 \cdot 10^0$		

Table 9.4

The results based on load pattern III are presented in tables 9.5 and 9.6. Here we increase the three packet streams simultaneously by one packet at a time. As the three links have different capacities, the loads on the three links do not increase by the same factor. Again, the SMA policy does very well when ρ_1 is small (table 9.5). However, in table 9.6 we see that when $\lambda_1 = 26$, $\lambda_2 = 30$ and $\lambda_3 = 16$ ($\Rightarrow \rho_1 = 0.26$, $\rho_2 = 0.60$, $\rho_3 = 0.80$) B_1 is decreased by a factor 10 when p_1 increases from 1 to 6 and meanwhile the total number of blocked packets increases from $2.14 \cdot 10^{-1}$ to $2.64 \cdot 10^{-1}$.

$\lambda =$ ($\lambda_1, \lambda_2, \lambda_3$)	$\rho =$ (ρ_1, ρ_2, ρ_3)	P_1	$\underline{m}$	B_1	B_2	B_3	# lost packets	$\frac{m=0}{\# \text{lost}}$ packets	$\frac{m=0}{B_1=B_2=B_3}$
21,25,11	0.021,0.50,0.55	1	0,0,0	$1.40 \cdot 10^{-5}$	$1.40 \cdot 10^{-5}$	$1.40 \cdot 10^{-5}$	$7.96 \cdot 10^{-4}$	$7.96 \cdot 10^{-4}$	$1.40 \cdot 10^{-5}$
		2..6	1,1,0	$5.71 \cdot 10^{-7}$	$2.20 \cdot 10^{-5}$	$2.67 \cdot 10^{-5}$	$8.55 \cdot 10^{-4}$		
22,26,12	0.022,0.52,0.60	1..2	1,2,0	$2.46 \cdot 10^{-6}$	$6.09 \cdot 10^{-5}$	$1.10 \cdot 10^{-4}$	$2.96 \cdot 10^{-3}$	$3.05 \cdot 10^{-3}$	$5.08 \cdot 10^{-5}$
		2..6	1,1,0	$2.02 \cdot 10^{-6}$	$1.11 \cdot 10^{-5}$	$9.06 \cdot 10^{-5}$	$2.95 \cdot 10^{-3}$		
23,27,13	0.023,0.54,0.65	1..6	1,2,0	$7.93 \cdot 10^{-6}$	$1.77 \cdot 10^{-4}$	$3.41 \cdot 10^{-4}$	$9.38 \cdot 10^{-3}$	$1.08 \cdot 10^{-2}$	$1.71 \cdot 10^{-4}$
24,28,14	0.024,0.56,0.70	1..6	1,2,0	$2.31 \cdot 10^{-5}$	$4.74 \cdot 10^{-4}$	$9.54 \cdot 10^{-4}$	$2.72 \cdot 10^{-2}$	$3.48 \cdot 10^{-2}$	$5.28 \cdot 10^{-4}$
25,29,15	0.025,0.58,0.75	1..6	1,2,0	$6.11 \cdot 10^{-5}$	$1.17 \cdot 10^{-3}$	$2.43 \cdot 10^{-3}$	$7.18 \cdot 10^{-2}$	$1.02 \cdot 10^{-1}$	$1.48 \cdot 10^{-3}$
26,30,16	0.026,0.60,0.80	1..6	1,3,0	$1.70 \cdot 10^{-4}$	$1.98 \cdot 10^{-3}$	$6.51 \cdot 10^{-3}$	$1.68 \cdot 10^{-1}$	$2.71 \cdot 10^{-1}$	$3.76 \cdot 10^{-3}$
27,31,17	0.027,0.62,0.85	1..6	1,3,0	$3.35 \cdot 10^{-4}$	$3.99 \cdot 10^{-3}$	$1.31 \cdot 10^{-2}$	$3.56 \cdot 10^{-1}$	$1.41 \cdot 10^{-1}$	$2.55 \cdot 10^{-3}$
28,32,18	0.028,0.64,0.90	1..6	1,4,0	$7.46 \cdot 10^{-4}$	$5.31 \cdot 10^{-3}$	$2.65 \cdot 10^{-2}$	$6.69 \cdot 10^{-1}$	$1.37 \cdot 10^0$	$1.76 \cdot 10^{-2}$
29,33,19	0.029,0.66,0.95	1	1,5,1	$1.35 \cdot 10^{-3}$	$6.40 \cdot 10^{-3}$	$4.66 \cdot 10^{-2}$	$1.14 \cdot 10^0$	$2.61 \cdot 10^0$	$3.22 \cdot 10^{-2}$
		2	1,5,0	$1.35 \cdot 10^{-3}$	$6.39 \cdot 10^{-3}$	$4.66 \cdot 10^{-2}$	$1.14 \cdot 10^0$		
		3..6	2,2,1	$4.34 \cdot 10^{-4}$	$1.15 \cdot 10^{-2}$	$5.14 \cdot 10^{-2}$	$1.14 \cdot 10^0$		
31,35,21	0.031,0.70,1.05	1	1,6,1	$3.01 \cdot 10^{-3}$	$1.04 \cdot 10^{-2}$	$9.73 \cdot 10^{-2}$	$2.50 \cdot 10^0$	$6.90 \cdot 10^0$	$7.93 \cdot 10^{-2}$
		2..6	2,6,1	$9.86 \cdot 10^{-5}$	$1.10 \cdot 10^{-2}$	$1.03 \cdot 10^{-1}$	$2.54 \cdot 10^0$		
32,36,22	0.032,0.72,1.10	1	1,7,1	$4.11 \cdot 10^{-3}$	$1.07 \cdot 10^{-2}$	$1.29 \cdot 10^{-1}$	$3.35 \cdot 10^0$	$9.82 \cdot 10^0$	$1.09 \cdot 10^{-1}$
		2..6	2,7,1	$1.35 \cdot 10^{-4}$	$1.10 \cdot 10^{-2}$	$1.10 \cdot 10^{-1}$	$3.36 \cdot 10^0$		
33,37,23	0.033,0.74,1.15	1..6	2,8,1	$1.81 \cdot 10^{-4}$	$1.12 \cdot 10^{-2}$	$1.67 \cdot 10^{-1}$	$4.26 \cdot 10^0$	$1.31 \cdot 10^1$	$1.41 \cdot 10^{-1}$
34,38,24	0.034,0.76,1.20	1..6	2,8,1	$2.23 \cdot 10^{-4}$	$1.46 \cdot 10^{-2}$	$1.94 \cdot 10^{-1}$	$5.21 \cdot 10^0$	$1.66 \cdot 10^1$	$1.72 \cdot 10^{-1}$
35,39,25	0.035,0.78,1.25	1..6	2,9,1	$2.74 \cdot 10^{-4}$	$1.46 \cdot 10^{-2}$	$2.25 \cdot 10^{-1}$	$6.21 \cdot 10^0$	$2.01 \cdot 10^1$	$2.03 \cdot 10^{-1}$
36,40,26	0.036,0.80,1.30	1..6	2,9,1	$3.21 \cdot 10^{-4}$	$1.85 \cdot 10^{-2}$	$2.50 \cdot 10^{-1}$	$7.25 \cdot 10^0$	$2.37 \cdot 10^1$	$2.33 \cdot 10^{-1}$
37,41,27	0.037,0.82,1.35	1..6	2,10,1	$3.78 \cdot 10^{-4}$	$1.88 \cdot 10^{-2}$	$2.79 \cdot 10^{-1}$	$8.32 \cdot 10^0$	$2.73 \cdot 10^1$	$2.60 \cdot 10^{-1}$
38,42,28	0.038,0.84,1.40	1..6	2,10,1	$4.31 \cdot 10^{-4}$	$2.34 \cdot 10^{-2}$	$3.01 \cdot 10^{-1}$	$9.44 \cdot 10^0$	$3.09 \cdot 10^1$	$2.86 \cdot 10^{-1}$
39,43,29	0.039,0.86,1.45	1..6	2,10,1	$4.94 \cdot 10^{-4}$	$2.43 \cdot 10^{-2}$	$3.28 \cdot 10^{-1}$	$1.06 \cdot 10^1$	$3.45 \cdot 10^1$	$3.11 \cdot 10^{-1}$
40,44,30	0.040,0.88,1.50	1..6	2,11,2	$5.51 \cdot 10^{-4}$	$2.98 \cdot 10^{-2}$	$3.48 \cdot 10^{-1}$	$1.18 \cdot 10^1$	$3.80 \cdot 10^1$	$3.34 \cdot 10^{-1}$
41,45,31	0.041,0.90,1.55	1..6	2,11,2	$6.09 \cdot 10^{-4}$	$3.59 \cdot 10^{-2}$	$3.67 \cdot 10^{-1}$	$1.30 \cdot 10^1$	$4.15 \cdot 10^1$	$3.55 \cdot 10^{-1}$
42,46,32	0.042,0.92,1.60	1..6	2,12,2	$6.81 \cdot 10^{-4}$	$3.82 \cdot 10^{-2}$	$3.91 \cdot 10^{-1}$	$1.43 \cdot 10^1$	$4.50 \cdot 10^1$	$3.75 \cdot 10^{-1}$
43,47,33	0.043,0.94,1.65	1..6	2,12,2	$7.44 \cdot 10^{-4}$	$4.54 \cdot 10^{-2}$	$4.08 \cdot 10^{-1}$	$1.56 \cdot 10^1$	$4.85 \cdot 10^1$	$3.94 \cdot 10^{-1}$
44,48,34	0.044,0.96,1.70	1..6	2,12,2	$8.09 \cdot 10^{-4}$	$5.33 \cdot 10^{-2}$	$4.23 \cdot 10^{-1}$	$1.44 \cdot 10^1$	$5.19 \cdot 10^1$	$4.12 \cdot 10^{-1}$
45,49,35	0.045,0.98,1.75	1..6	2,12,2	$8.76 \cdot 10^{-4}$	$6.18 \cdot 10^{-2}$	$4.39 \cdot 10^{-1}$	$1.84 \cdot 10^1$	$5.53 \cdot 10^1$	$4.29 \cdot 10^{-1}$

Table 9.5

$(\lambda_1, \lambda_2, \lambda_3)$	$\rho = (\rho_1, \rho_2, \rho_3)$	P_1	$\underline{m}$	B_1	B_2	B_3	# lost packets	$\underline{m} = 0$ # lost packets	$\underline{m} = 0$ $B_1 = B_2 = B_3$
21,25,11	0.21,0.50,0.55	1	0,0,0	$1.74 \cdot 10^{-5}$	$1.74 \cdot 10^{-5}$	$1.74 \cdot 10^{-5}$	$9.90 \cdot 10^{-4}$	$9.90 \cdot 10^{-4}$	$1.74 \cdot 10^{-5}$
		2..3	1,1,0	$7.10 \cdot 10^{-6}$	$2.32 \cdot 10^{-5}$	$2.82 \cdot 10^{-5}$	$1.04 \cdot 10^{-3}$		
		4..6	1,0,0	$6.52 \cdot 10^{-6}$	$2.59 \cdot 10^{-5}$	$2.59 \cdot 10^{-5}$	$1.07 \cdot 10^{-3}$		
22,26,12	0.22,0.52,0.60	1..4	1,1,0	$2.44 \cdot 10^{-5}$	$7.45 \cdot 10^{-5}$	$9.50 \cdot 10^{-5}$	$3.61 \cdot 10^{-3}$	$3.68 \cdot 10^{-3}$	$6.13 \cdot 10^{-5}$
		5	1,0,0	$2.29 \cdot 10^{-5}$	$8.68 \cdot 10^{-5}$	$8.68 \cdot 10^{-5}$	$3.79 \cdot 10^{-3}$		
		6	2,1,0	$8.86 \cdot 10^{-6}$	$1.18 \cdot 10^{-4}$	$1.51 \cdot 10^{-4}$	$5.09 \cdot 10^{-3}$		
23,27,13	0.23,0.54,0.65	1..4	1,1,0	$7.74 \cdot 10^{-5}$	$2.23 \cdot 10^{-4}$	$2.95 \cdot 10^{-4}$	$1.16 \cdot 10^{-2}$	$1.27 \cdot 10^{-2}$	$2.01 \cdot 10^{-4}$
		5..6	2,1,0	$2.72 \cdot 10^{-5}$	$3.30 \cdot 10^{-4}$	$4.38 \cdot 10^{-4}$	$1.52 \cdot 10^{-2}$		
24,28,14	0.24,0.56,0.70	1	1,2,0	$2.64 \cdot 10^{-4}$	$4.91 \cdot 10^{-4}$	$9.90 \cdot 10^{-4}$	$3.40 \cdot 10^{-2}$	$3.98 \cdot 10^{-2}$	$6.03 \cdot 10^{-4}$
		2	1,1,0	$2.24 \cdot 10^{-4}$	$6.16 \cdot 10^{-4}$	$8.40 \cdot 10^{-4}$	$3.44 \cdot 10^{-2}$		
		3..5	2,2,0	$9.05 \cdot 10^{-5}$	$6.80 \cdot 10^{-4}$	$1.37 \cdot 10^{-3}$	$4.04 \cdot 10^{-2}$		
		6	2,1,0	$7.68 \cdot 10^{-5}$	$8.54 \cdot 10^{-4}$	$1.16 \cdot 10^{-3}$	$4.21 \cdot 10^{-2}$		
25,29,15	0.25,0.58,0.75	1	1,2,0	$6.81 \cdot 10^{-4}$	$1.20 \cdot 10^{-3}$	$2.50 \cdot 10^{-3}$	$8.95 \cdot 10^{-2}$	$1.14 \cdot 10^{-1}$	$1.65 \cdot 10^{-3}$
		2..6	2,2,0	$2.27 \cdot 10^{-4}$	$1.57 \cdot 10^{-3}$	$3.26 \cdot 10^{-3}$	$1.00 \cdot 10^{-1}$		
26,30,16	0.26,0.60,0.80	1	1,2,0	$1.59 \cdot 10^{-3}$	$2.70 \cdot 10^{-3}$	$5.73 \cdot 10^{-3}$	$2.14 \cdot 10^{-1}$	$2.95 \cdot 10^{-1}$	$4.10 \cdot 10^{-3}$
		2	2,3,0	$6.06 \cdot 10^{-4}$	$2.50 \cdot 10^{-3}$	$8.24 \cdot 10^{-3}$	$2.23 \cdot 10^{-1}$		
		3..5	2,2,0	$5.19 \cdot 10^{-4}$	$3.33 \cdot 10^{-3}$	$7.06 \cdot 10^{-3}$	$2.26 \cdot 10^{-1}$		
		6	3,3,0	$1.98 \cdot 10^{-4}$	$3.13 \cdot 10^{-3}$	$1.03 \cdot 10^{-2}$	$2.64 \cdot 10^{-1}$		
27,31,17	0.27,0.62,0.85	1..3	2,3,0	$1.22 \cdot 10^{-3}$	$4.77 \cdot 10^{-3}$	$1.57 \cdot 10^{-2}$	$4.47 \cdot 10^{-1}$	$6.88 \cdot 10^{-1}$	$9.17 \cdot 10^{-3}$
		4..6	3,3,0	$3.92 \cdot 10^{-4}$	$5.66 \cdot 10^{-3}$	$1.87 \cdot 10^{-2}$	$5.04 \cdot 10^{-1}$		
28,32,18	0.28,0.64,0.90	1..2	2,4,0	$2.48 \cdot 10^{-3}$	$6.08 \cdot 10^{-3}$	$3.04 \cdot 10^{-2}$	$8.12 \cdot 10^{-1}$	$1.44 \cdot 10^0$	$1.84 \cdot 10^{-2}$
		3..6	3,4,0	$7.96 \cdot 10^{-4}$	$6.93 \cdot 10^{-3}$	$3.47 \cdot 10^{-2}$	$8.69 \cdot 10^{-1}$		
29,33,19	0.29,0.66,0.95	1	2,4,0	$4.07 \cdot 10^{-3}$	$9.78 \cdot 10^{-3}$	$4.74 \cdot 10^{-2}$	$1.34 \cdot 10^0$	$2.70 \cdot 10^0$	$3.33 \cdot 10^{-2}$
		2..5	3,4,0	$1.30 \cdot 10^{-3}$	$1.08 \cdot 10^{-2}$	$5.22 \cdot 10^{-2}$	$1.39 \cdot 10^0$		
		6	4,4,0	$4.19 \cdot 10^{-4}$	$1.19 \cdot 10^{-2}$	$5.79 \cdot 10^{-2}$	$1.51 \cdot 10^0$		
31,35,21	0.31,0.70,1.05	1..2	3,6,1	$3.18 \cdot 10^{-3}$	$1.16 \cdot 10^{-2}$	$1.09 \cdot 10^{-1}$	$2.79 \cdot 10^0$	$6.99 \cdot 10^0$	$8.03 \cdot 10^{-2}$
		3..6	4,6,1	$1.05 \cdot 10^{-3}$	$1.24 \cdot 10^{-2}$	$1.16 \cdot 10^{-1}$	$2.41 \cdot 10^0$		
32,36,22	0.32,0.72,1.10	1	3,6,1	$4.25 \cdot 10^{-3}$	$1.56 \cdot 10^{-2}$	$1.35 \cdot 10^{-1}$	$3.67 \cdot 10^0$	$9.90 \cdot 10^0$	$1.10 \cdot 10^{-1}$
		2..6	4,6,1	$1.43 \cdot 10^{-3}$	$1.63 \cdot 10^{-2}$	$1.42 \cdot 10^{-1}$	$3.75 \cdot 10^0$		
33,37,23	0.33,0.74,1.15	1	3,7,1	$5.67 \cdot 10^{-3}$	$1.53 \cdot 10^{-2}$	$1.68 \cdot 10^{-1}$	$4.61 \cdot 10^0$	$1.32 \cdot 10^1$	$1.41 \cdot 10^{-1}$
		2..5	4,7,1	$1.95 \cdot 10^{-3}$	$1.59 \cdot 10^{-2}$	$1.75 \cdot 10^{-1}$	$4.67 \cdot 10^0$		
		6	5,6,1	$6.46 \cdot 10^{-4}$	$2.18 \cdot 10^{-2}$	$1.75 \cdot 10^{-1}$	$4.86 \cdot 10^0$		
34,38,24	0.34,0.76,1.20	1	3,8,1	$7.27 \cdot 10^{-3}$	$1.51 \cdot 10^{-2}$	$2.00 \cdot 10^{-1}$	$5.62 \cdot 10^0$	$1.66 \cdot 10^1$	$1.73 \cdot 10^{-1}$
		2..4	4,7,1	$2.47 \cdot 10^{-3}$	$2.01 \cdot 10^{-2}$	$2.00 \cdot 10^{-1}$	$5.66 \cdot 10^0$		
		5..6	5,7,1	$8.74 \cdot 10^{-4}$	$2.08 \cdot 10^{-2}$	$2.09 \cdot 10^{-1}$	$5.83 \cdot 10^0$		

Table 9.6

$\lambda =$ ($\lambda_1, \lambda_2, \lambda_3$)	$\rho =$ (ρ_1, ρ_2, ρ_3)	P_1	$\underline{m}$	B_1	B_2	B_3	# lost packets	$\underline{m} = 0$ # lost packets	$\underline{m} = 0$ $B_1 = B_2 = B_3$
35, 39, 25	0.35, 0.78, 1.25	1	3, 8, 1	$8.77 \cdot 10^{-3}$	$1.90 \cdot 10^{-2}$	$2.25 \cdot 10^{-1}$	$6.68 \cdot 10^0$	$2.02 \cdot 10^1$	$2.04 \cdot 10^{-1}$
		2..3	4, 8, 1	$3.16 \cdot 10^{-3}$	$1.96 \cdot 10^{-2}$	$2.33 \cdot 10^{-1}$	$6.69 \cdot 10^0$		
		4..6	5, 7, 1	$1.11 \cdot 10^{-3}$	$2.56 \cdot 10^{-2}$	$2.33 \cdot 10^{-1}$	$6.86 \cdot 10^0$		
36, 40, 26	0.36, 0.80, 1.30	1..2	4, 8, 1	$3.82 \cdot 10^{-3}$	$2.41 \cdot 10^{-2}$	$2.56 \cdot 10^{-1}$	$7.77 \cdot 10^0$	$2.38 \cdot 10^1$	$2.33 \cdot 10^{-1}$
		3..6	5, 8, 1	$1.42 \cdot 10^{-3}$	$2.48 \cdot 10^{-2}$	$2.65 \cdot 10^{-1}$	$7.93 \cdot 10^0$		
37, 41, 27	0.37, 0.82, 1.35	1..2	4, 9, 1	$4.68 \cdot 10^{-3}$	$2.39 \cdot 10^{-2}$	$2.87 \cdot 10^{-1}$	$8.90 \cdot 10^0$	$2.34 \cdot 10^1$	$2.60 \cdot 10^{-1}$
		3..6	5, 8, 1	$1.73 \cdot 10^{-3}$	$2.99 \cdot 10^{-2}$	$2.87 \cdot 10^{-1}$	$9.04 \cdot 10^0$		
38, 42, 28	0.38, 0.84, 1.40	1..2	4, 9, 1	$5.49 \cdot 10^{-3}$	$2.90 \cdot 10^{-2}$	$3.08 \cdot 10^{-1}$	$1.01 \cdot 10^1$	$3.09 \cdot 10^1$	$2.86 \cdot 10^{-1}$
		3..6	5, 8, 1	$2.09 \cdot 10^{-3}$	$3.55 \cdot 10^{-2}$	$3.09 \cdot 10^{-1}$	$1.02 \cdot 10^1$		
39, 43, 29	0.39, 0.86, 1.45	1	4, 9, 1	$6.38 \cdot 10^{-3}$	$3.46 \cdot 10^{-2}$	$3.29 \cdot 10^{-1}$	$1.13 \cdot 10^1$	$3.45 \cdot 10^1$	$3.11 \cdot 10^{-1}$
		2..5	5, 9, 1	$2.55 \cdot 10^{-3}$	$3.53 \cdot 10^{-2}$	$3.38 \cdot 10^{-1}$	$1.14 \cdot 10^1$		
		6	6, 8, 1	$9.95 \cdot 10^{-4}$	$4.25 \cdot 10^{-2}$	$3.38 \cdot 10^{-1}$	$1.17 \cdot 10^1$		
40, 44, 30	0.40, 0.88, 1.50	1	4, 10, 1	$7.51 \cdot 10^{-3}$	$3.53 \cdot 10^{-2}$	$3.56 \cdot 10^{-1}$	$1.25 \cdot 10^1$	$3.80 \cdot 10^1$	$3.34 \cdot 10^{-1}$
		2..4	5, 9, 1	$3.00 \cdot 10^{-3}$	$4.15 \cdot 10^{-2}$	$3.56 \cdot 10^{-1}$	$1.26 \cdot 10^1$		
		5..6	6, 8, 1	$1.20 \cdot 10^{-3}$	$4.92 \cdot 10^{-2}$	$3.57 \cdot 10^{-1}$	$1.29 \cdot 10^1$		
41, 45, 31	0.41, 0.90, 1.55	1	4, 10, 1	$8.55 \cdot 10^{-3}$	$4.17 \cdot 10^{-2}$	$3.74 \cdot 10^{-1}$	$1.38 \cdot 10^1$	$4.15 \cdot 10^1$	$3.55 \cdot 10^{-1}$
		2..4	5, 9, 1	$3.50 \cdot 10^{-3}$	$4.83 \cdot 10^{-2}$	$3.75 \cdot 10^{-1}$	$1.39 \cdot 10^1$		
		5..6	6, 8, 1	$1.43 \cdot 10^{-3}$	$5.64 \cdot 10^{-2}$	$3.75 \cdot 10^{-1}$	$1.42 \cdot 10^1$		
42, 46, 32	0.42, 0.92, 1.60	1	4, 10, 2	$9.71 \cdot 10^{-3}$	$4.90 \cdot 10^{-2}$	$3.91 \cdot 10^{-1}$	$1.52 \cdot 10^1$	$4.50 \cdot 10^1$	$3.75 \cdot 10^{-1}$
		2..3	5, 9, 1	$4.05 \cdot 10^{-3}$	$5.57 \cdot 10^{-2}$	$3.92 \cdot 10^{-1}$	$1.53 \cdot 10^1$		
		4..6	6, 9, 1	$1.74 \cdot 10^{-3}$	$5.65 \cdot 10^{-2}$	$4.02 \cdot 10^{-1}$	$1.55 \cdot 10^1$		
43, 47, 33	0.43, 0.94, 1.65	1	4, 10, 2	$1.09 \cdot 10^{-2}$	$5.67 \cdot 10^{-2}$	$4.08 \cdot 10^{-1}$	$1.66 \cdot 10^1$	$4.85 \cdot 10^1$	$3.94 \cdot 10^{-1}$
		2	5, 10, 2	$4.62 \cdot 10^{-3}$	$5.75 \cdot 10^{-2}$	$4.17 \cdot 10^{-1}$	$1.66 \cdot 10^1$		
		3	5, 10, 1	$4.76 \cdot 10^{-3}$	$5.70 \cdot 10^{-2}$	$4.18 \cdot 10^{-1}$	$1.67 \cdot 10^1$		
		4..6	6, 9, 1	$2.05 \cdot 10^{-3}$	$6.43 \cdot 10^{-2}$	$4.18 \cdot 10^{-1}$	$1.69 \cdot 10^1$		
44, 48, 34	0.44, 0.96, 1.70	1	4, 10, 2	$1.22 \cdot 10^{-2}$	$6.49 \cdot 10^{-2}$	$4.24 \cdot 10^{-1}$	$1.81 \cdot 10^1$	$5.19 \cdot 10^1$	$4.12 \cdot 10^{-1}$
		2	5, 10, 2	$5.49 \cdot 10^{-3}$	$6.57 \cdot 10^{-2}$	$4.32 \cdot 10^{-1}$	$1.81 \cdot 10^1$		
		3..5	6, 9, 2	$2.41 \cdot 10^{-3}$	$7.32 \cdot 10^{-2}$	$4.32 \cdot 10^{-1}$	$1.83 \cdot 10^1$		
		6	6, 9, 1	$2.39 \cdot 10^{-3}$	$7.26 \cdot 10^{-2}$	$4.33 \cdot 10^{-1}$	$1.83 \cdot 10^1$		
45, 49, 35	0.45, 0.98, 1.75	1..2	5, 10, 2	$5.22 \cdot 10^{-3}$	$7.44 \cdot 10^{-2}$	$4.46 \cdot 10^{-1}$	$1.96 \cdot 10^1$	$5.53 \cdot 10^1$	$4.29 \cdot 10^{-1}$
		3..6	6, 9, 2	$2.80 \cdot 10^{-3}$	$8.20 \cdot 10^{-2}$	$4.47 \cdot 10^{-1}$	$1.98 \cdot 10^1$		

Table 9.6, cont.

9.4 Conclusions

As seen from our results we can, by introducing weight factors in our cost function, decrease blocking probabilities on some of the links leaving a node. Of course, this leads to an increase of blocking probabilities on the rest of the outgoing links.

We could express a desirable policy as follows:

If a packet is lost on its way from an origin host to a destination host, it is best to drop it as early as possible.

This implies that the value of the blocking probabilities on the different links in a path should get lower and lower as our packet proceeds through the net. We can hardly fulfil this policy completely, but the proposed cost function with weight factors forms one step towards the ideal policy.

Chapter 10 - A MODEL OF A PACKET SWITCHING NETWORK

10.1 Introduction

In this chapter we will incorporate our packet switching node model, presented in chapter 6, into a global packet switching network model, i e we develop a finite storage model for the subnetwork of an entire network. The finite storage model will incorporate our restricted buffer sharing policies and yield not only results in terms of values of different performance measures, but also an understanding of the behaviour of our packet switching network.

In the early and pioneering works of Kleinrock [53], where he assumed: external Poisson arrivals, exponential packet length distribution, error free channels, no nodal delay, independence, fix routing and infinite storage, the packet switching computer communication network was modelled as a queueing network [43]. This brought about that each link, and the queue associated to that link, behaved as an independent M/M/1 system, and that performance measures could easily be obtained by the known results from both the queueing network theory and the theory for M/M/1 systems.

In this chapter we will use the above listed assumptions except the one regarding the buffer space in the nodes.

After having presented our basic assumptions in 10.2 we present in 10.3 two sample networks to which we will apply our models and algorithms. These two networks will be used in 10.6, where we present results in terms of network utilization or network throughput, the number of

lost packets per second, delays etc. In 10.4 we derive different algorithms to compute the different link flows, first in 10.4.1, when the link blocking probabilities are known, and then in 10.4.2, when they are not known. We close section 10.4 with a discussion on the convergence of our algorithms and propose a slightly modified algorithm for the case when blocking probabilities are unknown. In 10.5 we present algorithms to compute end to end blocking probabilities and delays and discuss how to use Little's formula in a network with loss. In section 10.6 we apply our earlier presented algorithms to the two sample networks in 10.3, one 6-node - and one 11-node network. In 10.7 we finally make conclusions and present a review of earlier network analyses.

10.2 Basic assumptions

Our model of a packet switching computer communication network incorporates only the sub network, i e the nodes and the transmission links between the nodes. Thus the hosts and the transmission links between the hosts and the nodes are omitted in our model. This implies that packets sent from one host to another can be seen generated in the node to which the sending host is linked and delivered to the node to which the receiving host is linked. The performance measures will therefore be restricted to those within the sub network.

We assume packets being generated in node s and destined for node t to form a Poisson process with the intensity $\text{GAMMA}(s,t) \forall s,t = 1,2,\dots,n$.

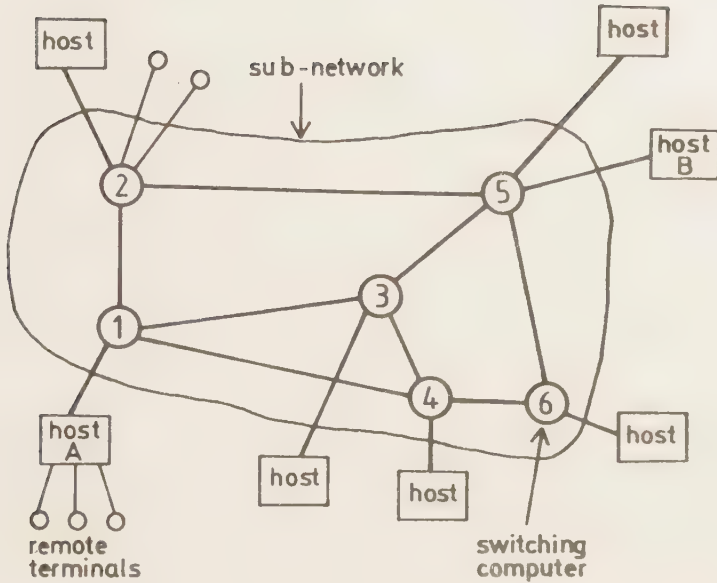


Figure 10.1

Furthermore we assume packet lengths to be exponentially distributed with mean $1/\mu$ bits. We use Kleinrock's independence assumption, i.e. a new packet length is drawn from the exponential distribution for every new transmission in the network.

This independence assumption is extended to the general independence assumption [52], i.e. for every node in the network j -packet arrivals are assumed to form a Poisson process. This leads to that the nodes can be regarded independent of each other.

As mentioned above Burke [9] has shown the output process from a M/M/1 system to be a Poisson process. But

we know, however, that the output process from an "M/M/1. limited number of queue places" is not Poisson [27],[83].

However, the effects of the general independence assumption have been thoroughly studied by comparisons between simulations with and without that assumption and could briefly be summarized as follows: The effects are not crucial for meshed networks, i.e. networks where every node is linked to several other nodes. But the general independence assumption is generally inadequate for tandem networks [53],[29],[69].

Furthermore, in chapter 4 (4.6.4) we briefly discussed the effects of merging and splitting renewal processes and found that both operations tend to create Poissonian behaviour.

10.3 Sample Networks

We have chosen to model two different networks, consisting of the sub network only, that is



Figure 10.2

only the packet nodes and the links between the nodes. Thus blocking probabilities and delays between host and node and vice versa are not included in the corresponding performance measures between source and destination hosts. All links are fully duplex. The topology of the first net, a 6-node net, is found in figure 10.2.

The corresponding capacity matrix CAP shows the different link capacities in kbits/sec.

	0	15	15	0	0	0
	15	0	10	20	0	0
CAP =	15	10	0	0	20	0
	0	20	0	0	15	15
	0	0	20	15	0	10
	0	0	0	15	10	0

Table 10.1

Furthermore, the two networks use a fix routing algorithm. In table 10.2 we find the routing matrix ROUT for the 6-node net, where an element $ROUT(i,j)$ indicates toward which node a packet in node i , bound for destination node j , is to be sent.

	0	2	3	2	3	2
	1	0	3	4	4	4
ROUT =	1	2	0	5	5	5
	2	2	5	0	5	6
	3	4	3	4	0	6
	4	4	5	4	5	0

Table 10.2

The second net, an 11-node net, has the following topology.

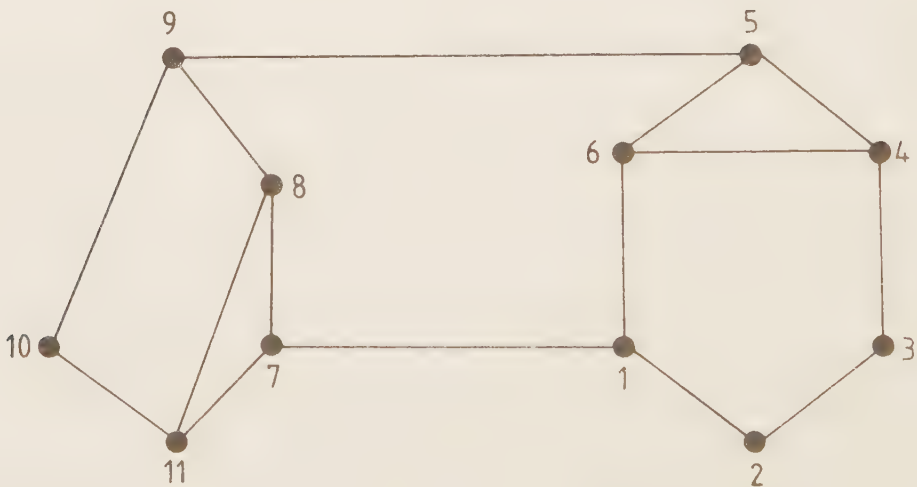


Figure 10.3

The corresponding capacity matrix CAP is given in table 10.3.

	0	20	0	0	0	100	100	0	0	0	0
	20	0	20	0	0	0	0	0	0	0	0
	0	20	0	50	0	0	0	0	0	0	0
	0	0	50	0	50	50	0	0	0	0	0
	0	0	0	50	0	100	0	0	50	0	0
CAP =	100	0	0	50	100	0	0	0	0	0	0
	100	0	0	0	0	0	0	50	0	0	20
	0	0	0	0	0	0	50	0	50	0	20
	0	0	0	0	50	0	0	50	0	20	0
	0	0	0	0	0	0	0	0	20	0	20
	0	0	0	0	0	0	20	20	0	20	0

Table 10.3

Finally the routing matrix ROUT is given in table 10.4.

	0	2	6	6	6	6	7	7	7	7	7
	1	0	3	3	3	3	1	1	1	1	1
	4	2	0	4	4	4	4	4	4	4	4
	6	3	3	0	5	5	6	6	5	5	6
	6	4	4	4	0	6	6	9	9	9	9
ROUT =	1	1	4	4	5	0	1	1	5	5	1
	1	1	1	1	1	1	0	8	8	11	11
	7	7	7	7	9	7	7	0	9	9	11
	8	8	5	5	5	5	8	8	0	10	8
	11	11	9	9	9	9	11	9	9	0	11
	7	7	7	7	8	7	7	8	8	10	0

Table 10.4

The first of these two networks, the 6 node net, should be regarded as a small graspable network. The topology of the second network, the 11 node net, has some similarities with that of the early ARPA net [38]. We find the nodes within the network being clustered in two areas (node 1-6 and node 7-11). These two areas are connected via the links between node 7 and node 1 and via node 9 and node 5.

The two nets are, regarding link capacities and routing, fully symmetrical, i.e. $C_{ij} = C_{ji} \forall i, j = 1, 2, \dots, n$, and a packet that is sent from node i to node j uses the same path (however in the other direction) as a packet sent from node j to node i . However, as we will see in section 10.6, Sample results, though the net being symmetrical even regarding offered arrival rate between any two nodes, i.e. $\gamma_{ij} = \gamma_{ji}$, load, utilization, throughput and delay on link i, j may not be identical to that on link j, i as packets are lost due to limited buffer storage.

10.4 The problem in essence

In order to compute the performance measures we are interested in, for instance end to end blocking probabilities and delays, we have to compute the different link flows, i.e. the number of j -packets offered node $i \forall i, j$. The link flows could be obtained by means of an iterative method using the source-destination packet intensity matrix, the link capacities and knowledge of the number of buffers and the restricted buffer sharing policies in the different nodes.

10.4.1 Algorithms to compute link flows when blocking is known

As we have assumed a limited fix routing policy, i.e.

a policy in which the choice of an outgoing link does only depend on the destination of the packet, we have that the next node for packets in node i , destined for node t , is given by the element $ROUT(i,t)$ in the two dimensional matrix $ROUT$.

Furthermore we give the intensity matrix $GAMMA$, where an element $GAMMA(s,t)$ is equal to the number of packets per second with origin node s and destined for node t .

Thus, if for the sake of simplicity we begin with a network with no loss, the link flow on link (i,j) , $LINKFLOW(i,j)$, is equal to the sum of those source-destination flows that use link (i,j) on their path between node s and t .

$$LINKFLOW(i,j) = \sum_s \sum_t GAMMA(s,t) \quad (10.1)$$

link (i,j) belongs
to path (s,t)

As a consequence of limited buffer space in the nodes, packets are lost on their path between origin node s and destination node t . When we compute $LINKFLOW$ under these new circumstances, we assume the blocking probabilities to be known. They are found in the matrix $PLOSS$. Now the source-destination rate $GAMMA(s,t)$ is reduced to $GAMMA(s,t) \cdot (1 - PATHLOSS(s,i))$, when it reaches node i (here we assume node i to be a node used in path (s,t)), where $PATHLOSS(s,i)$ is the end to end blocking probability between node s and node i . Thus we can write

$$LINKFLOW(i,j) = \sum_s \sum_t GAMMA(s,t) (1 - PATHLOSS(s,i)) \quad (10,2)$$

link (i,j) belongs
to path (s,t)

Below we present a fast algorithm to compute $LINKFLOW$.

```
program algorithm_1;
const noofnodes=20;
type nodenumber=1..noofnodes;
matrix1 = array [nodenumber,nodenumber]
           of nodenumber;
matrix2 = array [nodenumber,nodenumber] of real;
:
:
procedure algorithm_LINKFLOW(ROUT:matrix1;GAMMA,PLOSS:matrix2;
  var LINKFLOW:matrix2);
var source,destination,in_node,
      next_node,node_1,node_2 :nodenumber;
      pathnonloss :real;
begin for node_1:=1 to noofnodes do
      for node_2:=1 to noofnodes do
        LINKFLOW[node_1,node_2]:=0;
        for source:=1 to noofnodes do
          for destination:=1 to noofnodes do
            begin
              in_node:=source;
              next_node:=ROUT[in_node,destination];
              pathnonloss:=1;
              while in_node<>destination do
                begin
                  LINKFLOW[in_node,next_node]:=
                    LINKFLOW[in_node,next_node]+
                    GAMMA[source,destination]*pathnonloss;
                  pathnonloss:=
                    pathnonloss*(1-PLOSS[in_node,destination]);
                  in_node:=next_node;
                  next_node:=ROUT[in_node,destination];
                end
              end
            end;
          :
          :
        end;
      :
      :
    end;
    :
    :
```

The technique used allocates the flow produced by $\text{GAMMA}(s,t)$ to those elements in the matrix LINKFLOW , which together form path (s,t) and this procedure is repeated for all s and t . The algorithm is presented as a Pascal-like program.

10.4.2 An iterative algorithm to compute linkflow

Irland [40] has in his Ph D thesis proposed an algorithm to compute LINKFLOW when PLOSS is not given. In the previous section we computed LINKFLOW when PLOSS was given. Thus we can write LINKFLOW as a function FLOWS of PLOSS .

$$\text{LINKFLOW} = \text{FLOWS}(\text{PLOSS}) \quad (10.3)$$

The load matrix LOAD is obtained by normalizing the LINKFLOW with respect to the mean packet length in bits and to the link capacities given in the matrix CAP . Thus $\text{LOAD}(i,j) = \text{LINKFLOW}(i,j) \cdot \text{MEANPACKETLENGTH}/\text{CAP}(i,j)$ and we can write

$$\text{LOAD} = \text{NORM}(\text{LINKFLOW}) \quad (10.4)$$

To determine the blocking probabilities on the different links, we can view them as a function of the loads offered to the different nodes. The blocking probabilities do of course also depend on the number of buffers in the nodes and on a potential restricted buffer sharing policy, but anyhow, we can write

$$\text{PLOSS} = \text{LOSS}(\text{LOAD}) \quad (10.5)$$

and thus

$$\text{PLOSS} = \text{LOSS} * \text{NORM} * \text{FLOWS}(\text{PLOSS}) \quad (10.6)$$

where $*$ denotes superposition of transformations.

We will use a fixed-point iteration to solve this equation, with the initial estimate for all blocking probabilities being 0. That is

$$PLOSS_0 = \underline{0} \quad (10.7)$$

$$PLOSS_{p+1} = LOSS * NORM * FLOWS(PLOSS_p). \quad (10.8)$$

The iteration stops when $PLOSS_{p+1}$ is approximately equal to $PLOSS_p$.

Irland claims the stopping criterion for the iteration to be defined as

$$\frac{\sum_{i=1}^n \sum_{j=1}^n (PLOSS_{p+1}(i,j) - PLOSS_p(i,j))^2}{\sum_{i=1}^n \sum_{j=1}^n (PLOSS_{p+1}(i,j))^2} < \alpha \quad (10.9)$$

A tighter criterion for the iteration could be formulated as

$$\frac{(PLOSS_{p+1}(i,j) - PLOSS_p(i,j))^2}{PLOSS_{p+1}(i,j)^2} < \beta \quad (10.10)$$

$$\forall i, j = 1, 2, \dots, n;$$

This algorithm gives us the different linkflows and blocking probabilities and these entities are used in other algorithms to compute delay, utilization and blocking probabilities between any two nodes.

10.4.3 Convergence and length of iteration

As we are dealing with restricted buffer sharing algorithms in the different nodes in general and with Sharing with Minimum Allocation (SMA) in particular,

the number of dedicated buffers allocated to the different links are not known. Thus, for every loop in the iteration, we have to minimize our cost function containing the number of lost packets per second with respect to the number of dedicated buffers. However, if we do so, the coefficients in the equation $PLOSS_{p+1} = LOSS * NORM * FLOWS(PLOSS_p)$ are changed and we can hardly convince ourselves of convergence towards the global minimum. However, our experience strengthens our belief that the results from this method are very close to the global minimum.

When using this method on different networks, we have seen that the number of dedicated buffers seldom change after three loops in the iteration.

In order to strengthen our conviction of convergence towards a global minimum, we propose a slightly different method. Starting with $PLOSS_0 = 0$ we compute $PLOSS_1$ minimized over the number of allocated buffers and during the following loops in the iteration, we do not change the number of allocated buffers. The iteration uses one of the previous presented stopping criterions and the result from the iteration is used as a starting value for a new iteration, where a new number of allocated buffers is computed in the first loop. The total iteration, consisting of a number of subiterations, stops when the number of allocated buffers do not change between two subiterations. In this method the coefficients in the equation $PLOSS_{p+1} = LOSS * NORM * FLOWS(PLOSS_p)$ do not change within each subiteration. We also found this method much faster than the former.

We have computed linkflows, blocking probabilities and the number of dedicated buffers by means of the two different methods and have found no significant differ-

ences in our results. No differences whatsoever have been found for load up to a factor 4 (load pattern i) and a factor 15 (load pattern ii). See section 10.6, Sample results.) For higher loads we have found differences regarding net throughput of less than 0.1%.

10.5 Algorithms to compute end to end blocking probabilities and delays

The results of the algorithm used in section 10.4 to compute PLOSS and LINKFLOW could also give us a matrix NPACKL containing the mean number of i-packets in node j. A recursive technique is given in chapter 7.

In this section we will present an algorithm to compute

PATHLOSS: the "path blocking probabilities". An element PATHLOSS(s,t) is equal to the total blocking probabilities for packets originating in node s and destined for node t.

$$\text{Thus } \text{PATHLOSS}(s,t) = 1 - \pi(1 - \text{PLOSS}(i,j))_{\text{path}(s,t)}$$

The product is computed over those i and j where link(i,j) belongs to path(s,t).

PATHDELAY: an element PATHDELAY(s,t) is equal to the total delay for packets that originate in node s and that successfully reach destination node t.

$$\text{Thus } \text{PATHDELAY}(s,t) = \sum_{\text{path}(s,t)} \text{DELAY}(i,j)$$

where

$$\text{DELAY}(i,j) = \text{NPACKL}(i,j) / \text{LINKFLOW}(i,j) / (1 - \text{PLOSS}(i,j))$$

This last equation is simply Little's result $T = \bar{N}/\lambda$.

Furthermore, we wish to compute the meanpathloss and meanpathdelay for the entire network. We find

$$\text{meanpathloss} = \frac{1}{\text{gamma}} \cdot \sum_s \sum_t \text{GAMMA}(s,t) \cdot \text{PATHLOSS}(s,t); \quad (10.11)$$

and

$$\text{meanpathdelay} = \frac{1}{\text{gamma}(1 - \text{meanpathloss})} \cdot \sum_s \sum_t \text{GAMMA}(s,t) \cdot (1 - \text{PATHLOSS}(s,t)) \cdot \text{PATHDELAY}(s,t) \quad (10.12)$$

where

$$\text{gamma} = \sum_s \sum_t \text{GAMMA}(s,t) \quad (10.13)$$

The algorithm below produces, with the three matrices GAMMA, PLOSS and NPACKL as input, PATHLOSS, PATHDELAY, meanpathloss and meanpathdelay.

```
program    algorithm_2;  
const      noofnodes=20;  
type       nodenumber=1..noofnodes;  
matrix1     = array[nodenumber,nodenumber]  
              of nodenumber;  
matrix2     = array[nodenumber,nodenumber] of real;  
:  
:
```

```
procedure algorithm_PATH(ROUT:matrix1;GAMMA,NPACKL,PLOSS:
matrix2;gamma:real; var PATHLOSS,PATHDELAY:matrix2;
var meanpathdelay,meanpathloss:real);

var
source,destination,
in_node,next_node      :nodenumber;
sumpathloss,sumpathdelay,
sumploss,sumdelay      :real;

begin
sumpathloss:=0;
sumpathdelay:=0;
for source:=1 to noofnodes do
for destination:=1 to noofnodes do
begin sumploss:=1;
sumdelay:=0;
in_node:=source;
next_node:=ROUT[in_node,destination];
while in_node<>destination do
begin sumploss:=sumploss*(1-PLOSS[in_node,next_node]);
sumdelay:=sumdelay+NPACKL[in_node,next_node]/
(LINKFLOW[in_node,next_node]*
(1-PLOSS[in_node,next_node]));
in_node:=next_node;
next_node:=ROUT[in_node,destination];
end;
PATHLOSS[source,destination]:=1-sumploss;
PATHDELAY[source,destination]:=sumdelay;
sumpathloss:=sumpathloss+GAMMA[source,destination]*
PATHLOSS[source,destination];
sumpathdelay:=sumpathdelay+GAMMA[source,destination]*
(1-PATHLOSS[source,destination])*
PATHDELAY[source,destination];
end;
meanpathloss:=sumpathloss/gamma;
meanpathdelay:=sumpathdelay/(gamma*(1-meanpathloss));
end;

:
:
```

10.5.1 Matrix algorithms

In the previous section, 10.5, we used algorithm PATH to compute performance measures like PATHLOSS and PATH-DELAY. In this section we will present some "matrix-algorithms" used to compute the same measures as algorithm PATH, but these procedures are a bit more clarifying.

Besides the matrices defined in section 10.5 we will use the following matrices in our "matrix-algorithms"

IDN: the identity matrix of order $n \times n$

$$\text{IDN}(i,j) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

ZERO: the "zero-matrix" of order $n \times n$

$$\text{ZERO}(i,j) = 0 \quad \forall i,j = 1,2,\dots,n$$

ONEM: the "one-matrix" of order $n \times n$

$$\text{ONEM}(i,j) = 1 \quad \forall i,j = 1,2,\dots,n$$

ONEV: the "one-vector" of length n

$$\text{ONEV}(i) = 1 \quad \forall i = 1,2,\dots,n$$

A: any matrix of order $n \times n$ and with elements $a(i,j)$.

For every pair of nodes, s and t , in the network we define a matrix R_{st} , which describes the path in the network used by packets originating in node s and destined for node t . An element r_{ij} in R_{st} equals 1 if link (i,j) belongs to path (s,t) and is equal to 0 otherwise.

$$r_{ij} = \begin{cases} 1 & \text{if link } (i,j) \text{ belongs to path } (s,t) \\ 0 & \text{otherwise} \end{cases}$$

To compute R_{st} we use the following algorithm

```
program    algorithm_3;
const      noofnodes=20;
type       nodenumber=1..noofnodes;
            matrix1   = array [nodenumber,nodenumber]
                        of nodenumber;
            matrix3    = array [nodenumber,nodenumber] of 0..1;
            :
procedure  algorithm_Rst(ROUT:matrix1;source,
                        destination:nodenumber; var Rst:matrix3);
var        in_node,next_node,node_1,node_2:nodenumber;
begin
    for node_1:=1 to noofnodes do
    for node_2:=1 to noofnodes do
        Rst (node_1,node_2):=0;
    in_node:=source;
    next_node:=ROUT[source,destination];
    while in_node<>destination do
        begin Rst [in_node,next_node]:=1;
            in_node:=next_node;
            next_node:=ROUT[in_node,destination];
        end
    end;
    :
    :
```

An element (i,j) in the matrix (ONEM-PLOSS) is equal to the non-blocking probability on link (i,j) and an element in the matrix P_{st} where $P_{st}(i,j) = R_{st}(i,j) * (ONEM(i,j) - PLOSS(i,j))$ is equal to the non-blocking probability on link (i,j) if link (i,j) belongs to path (s,t) and 0 otherwise.

An element (i,j) in

$$Q_{st} = (ONEM - R_{st}) + P_{st} \quad (10.14)$$

is equal to the blocking probability on link (i,j) if link (i,j) belongs to path (s,t) and 1 otherwise.

Define an operator prod as

$$\text{prod}(A) = \prod_{i=1}^n \prod_{j=1}^n a(i,j) \quad (10.15)$$

Thus PATHLOSS(s,t) defined as the total blocking probability between node s and node t is given by

$$\text{PATHLOSS}(s,t) = 1 - \text{prod}(Q_{st}) \quad (10.16)$$

and the average blocking probability between any two nodes in the network as

$$\text{meanpathloss} = \sum_{s=1}^n \sum_{t=1}^n \frac{1}{\text{gamma}} \cdot \text{GAMMA}(s,t) \cdot \text{PATHLOSS}(s,t) \quad (10.17)$$

Let us define an operator sum as

$$\text{sum}(A) = \prod_{i=1}^n \prod_{j=1}^n a(i,j); \quad (10.18)$$

If we observe that

$$\text{sum}(A) = \text{ONEV} * A * \text{ONEV}^T \quad (10.19)$$

and form a matrix NOOFLOSTPATHPACK with elements

$$\text{NOOFLOSTPATHPACK}(s,t) = \text{GAMMA}(s,t) \cdot \text{PATHLOSS}(s,t) \quad (10.20)$$

then meanpathloss could be written

$$\text{meanpathloss} = \left(\frac{1}{\gamma}\right) \cdot \text{ONEV} * \text{NOOFLOSTPATHPACK} * \text{ONEV}^T \quad (10.21)$$

or shorter

$$\text{meanpathloss} = \left(\frac{1}{\gamma}\right) \cdot \text{sum}(\text{NOOFLOSTPATHPACK}) \quad (10.22)$$

Another way to compute $\text{PATHLOSS}(s,t)$ is obtained by defining a new matrix R'_{st} where the elements r'_{ij} are defined as

$$r'_{ij} = \begin{cases} 1 & \text{if link } (i,j) \text{ belongs to path } (s,t) \\ 1 & \text{if } i=j, i \neq t \text{ and link } (i,l) \text{ does not belong to path } (s,t) \text{ } l=1, \dots, n; \\ 1 & \text{if } i=t, j=s \\ 0 & \text{otherwise} \end{cases}$$

To compute R'_{st} we use an algorithm similar to the one used to compute R_{st} .


```
program    algorithm_4;

const      noofnodes=20;

type       nodenumber=1..noofnodes;
            matrix1    = array [nodenumber,nodenumber]
                        of nodenumber;
            matrix3     = array [nodenumber,nodenumber] of 0..1;

            :

procedure  algorithm_R'_st (ROUT:matrix1;source,
                        destination:nodenumber; var R'_st:matrix3);

var        in_node,next_node,node_1,node_2:nodenumber;

begin
    for node_1:=1 to noofnodes do
    for node_2:=1 to noofnodes do
    if node_1=node_2 then R'_st [node_1,node_2]:=1 else
                                R'_st [node_1,node_2]:=0,
    R'_st [destination,source]:=1;
    R'_st [destination,destination]:=0;
    in_node:=source;
    next_node:=ROUT[source,destination];
    while in_node<>destination do
        begin R'_st [in_node,next_node]:=1;
            R'_st [in_node,in_node]:=0;
            in_node:=next_node;
            next_node:=ROUT[in_node,destination];
        end
    end;

    :
    :
```

Let

$$Q'_{st}(i,j) = R'_{st}(i,j) \cdot (ONEM(i,j) - PLOSS(i,j)) \quad (10.23)$$

R'_{st} as well as Q'_{st} contains just one non-zero element in every row and just one non-zero element in every column.

Thus $\text{abs}(\det(Q'_{st}))$ is equal to the product of all non-zero elements in the matrix Q'_{st} and thus

$$\text{abs}(\det(Q'_{st})) = \text{prod}(Q_{st}) \quad (10.24)$$

and

$$\text{PATHLOSS}(s,t) = 1 - \text{abs}(\det(Q'_{st})). \quad (10.25)$$

10.5.2 Little's formula applied to a network with loss

In order to compute `meanpathdelay` we have

- i) computed the mean number of j -packets in node j using the recursive technique presented in chapter 7.
- ii) by means of Little's formula computed mean delay for j -packets in node i .
- iii) by means of algorithm `PATH` computed `PATHDELAY`, i.e. mean delay for packets originating in node s and destined for node t .
- iv) outgoing from `PATHDELAY` formed `meanpathdelay` according to eq 10.12

However, is it possible for us to apply Little's formula

$$T = \frac{\bar{N}}{\lambda_e} \quad (10.26)$$

directly on the complete network? Yes, of course! However, we have to be immensely careful regarding which measures that should be used in the formula.

When we talk about mean path delay, we mean the value taken over those packets being successfully delivered at their destination node. Thus the arrival rate used in Little's formula should be the one that corresponds to the number of successfully delivered packets and furthermore the mean number of packets in the network should be the mean number of packets in the network that will be successfully delivered.

Let us take an example: A two node tandem net for which we, in spite of our earlier discussions, assume independence between the two subsystems.



Figure 10.4

Mean delay for successfully delivered packets could of course be seen as the sum of mean delay in system 1 and that of system 2

$$T = T_1 + T_2 \quad (10.27)$$

Mean delay for successfully delivered packets in system 1, T_1 , is of course the same as mean delay for any packet in system 1 and equals T_1 where

$$T_1 = \frac{\bar{N}_1}{\lambda(1 - B_1)} \quad (10.28)$$

and where $\lambda(1 - B_1)$ is the effective arrival rate to system 1.

Furthermore, mean delay for successfully delivered packets in system 2, T_2 , equals

$$T_2 = \frac{\bar{N}_2}{\lambda(1 - B_1)(1 - B_2)} \quad (10.29)$$

where $\lambda(1 - B_1)(1 - B_2)$ is the effective arrival rate to system 2.

Thus

$$T = \frac{\bar{N}_1}{\lambda(1 - B_1)} + \frac{\bar{N}_2}{\lambda(1 - B_1)(1 - B_2)} \quad (10.30)$$

Let us now apply Little's formula directly on the complete tandem network

$$T = \frac{\bar{N}^1}{\lambda_e} \quad (10.31)$$

where N^1 is equal to the mean number of packets in the system that will be successfully delivered and λ_e is equal to the effective system arrival rate, i.e. the number of successfully delivered packets per second.

Thus

$$\bar{N}^1 = \bar{N}_1(1 - B_2) + \bar{N}_2 \quad (10.32)$$

and

$$\lambda_e = \lambda (1 - B_1) (1 - B_2) \quad (10.33)$$

Finally we get

$$T = \frac{\bar{N}_1 (1 - B_2) + \bar{N}_2}{\lambda (1 - B_1) (1 - B_2)} \quad (10.34)$$

$\Rightarrow$

$$T = \frac{\bar{N}_1}{\lambda (1 - B_1)} + \frac{\bar{N}_2}{\lambda (1 - B_1) (1 - B_2)} \quad (10.35)$$

Well, we found it not too hard to apply Little's formula to our two step tandem network. However, let us once more emphasize that we can of course use $\bar{N}$, i.e. the total number of packets in the network, in Little's formula to compute delay. If doing so we obtain mean delay for all packets that have entered the network. This mean value clearly includes the delays of blocked packets. Little's formula gives us

$$T = \frac{\bar{N}}{\lambda (1 - B_1)} \quad (10.36)$$

$\Rightarrow$

$$T = \frac{\bar{N}_1}{\lambda (1 - B_1)} + \frac{\bar{N}_2}{\lambda (1 - B_1)} \quad (10.37)$$

$\Rightarrow$

$$T = \frac{\bar{N}_1}{\lambda (1 - B_1)} + \frac{\bar{N}_2 (1 - B_2)}{\lambda (1 - B_1) (1 - B_2)} \quad (10.38)$$

$\Rightarrow$

$$T = T_1 + (1 - B_2) \cdot T_2 \quad (10.39)$$

All packets that have entered the network have on average a delay in the first subsystem of T_1 seconds. $(1 - B_2)$ of the packets having entered the network are not blocked in the second node and thus contribute with on average T_2 seconds to the total delay. The others,

the blocked ones (their proportion of the number having entered the network is B_2), do not contribute with a delay from the second subsystem.

Let us now apply these thoughts from our two step tandem network to a distributed meshed network.

How do we express the number of packets in the network that will be successfully delivered? Well, we first have to compute the mean number of packets in the network and then reduce it using the matrices GAMMA, LINK-FLOW and ROUT.

However, this will not reduce our computations in comparison with our earlier method, i e step i) to step iv).

10.6 Sample results

In this section we present different performance measures like throughput and delays based on two different load patterns

- i) all source-destination flows are simultaneously increased by the same factor.
- ii) one source-destination flow is increased and the others are held constant.

We compare our performance measures under our earlier presented restricted buffer sharing policies: CS, SMA, ESMA, FSMA, CP and FCP.

Furthermore we assume the mean packet lengths $\frac{1}{\mu}$ to be 1000 bits.

We have chosen to present more comprehensive results for the 11-node network than for the 6-node network.

10.6.1 The 6-node net

Under both load pattern i) and ii) we let the end to end arrival rates emanate from the following GAMMA matrix

	0	7	6	5	4	1
	7	0	7	7	2	4
GAMMA =	6	7	0	3	6	4
	5	7	3	0	6	7
	4	2	6	6	0	3
	1	4	4	7	3	0

Table 10.5

Under load pattern i) the intensity matrix GAMMA is increased by a factor γ and under load pattern ii) the intensity from node 2 to node 6, $\text{GAMMA}(2,6)$, is increased (multiplied by a factor called γ) and the other intensities are held constant.

Thus, when $\gamma = 1$ $\text{gamma} = \sum \sum \text{GAMMA}(s,t) = 144$ packets per second.

Furthermore we assume the total number of buffers in each node $M = 10$.

10.6.1.1 Offered load

Applying our link flow algorithms to this 6 node net, we find that offered loads on the different links, i e

$$\text{offered load } (i,j) = \frac{\text{linkflow}(i,j)}{\mu \cdot \text{CAP}(i,j)}$$

are given by the matrices in table 10.6 and table 10.7. Offered loads in the two tables are computed under the CS and the SMA policies respectively. In both tables offered loads are presented for $\gamma = 1$ and $\gamma = 3$ and under load pattern i).

0	0.87	0.67	0	0	0
0.85	0	0.70	0.95	0	0
0.66	0.70	0	0	0.85	0
0	0.95	0	0	0.73	0.78
0	0	0.84	0.73	0	0.69
0	0	0	0.80	0.70	0

a) $\gamma = 1$

0	2.60	2.00	0	0	0
1.88	0	2.10	2.30	0	0
1.60	2.10	0	0	2.18	0
0	2.26	0	0	1.97	1.77
0	0	1.95	1.86	0	1.43
0	0	0	2.40	2.10	0

b) $\gamma = 3$

offered load, CS

Table 10.6

0	0.87	0.67	0	0	0
0.85	0	0.70	0.95	0	0
0.66	0.70	0	0	0.85	0
0	0.95	0	0	0.73	0.78
0	0	0.84	0.73	0	0.69
0	0	0	0.80	0.70	0

a) $\gamma = 1$

0	2.60	2.00	0	0	0
1.88	0	2.10	2.30	0	0
1.60	2.10	0	0	2.25	0
0	2.27	0	0	1.97	1.78
0	0	2.01	1.87	0	1.43
0	0	0	2.40	2.10	0

b) $\gamma = 3$

offered load, SMA

Table 10.7

As seen, tables 10.6a and 10.7a are identical within two decimals. However, tables 10.6b and 10.7b do differ. It is important to notice that offered load is not identical to utilization of the corresponding link and that offered load must be identical for the CS and SMA policies (and for any policy) for those links (i,j) just carrying traffic originating in node i . When comparing offered load for the CS and the SMA policies we find the SMA policy to give a higher offered load.

10.6.1.2 The number of lost packets

Table 10.8a and b shows the number of lost packets per second on link (i,j). Actually the packets are not lost

0	24.02	18.48	0	0	0
16.30	0	12.15	26.57	0	0
13.48	11.77	0	0	24.45	0
0	25.61	0	0	16.73	15.05
0	0	19.74	14.18	0	7.24
0	0	0	21.15	12.34	0

a) CS

0	24.00	15.00	0	0	0
13.20	0	11.05	25.98	0	0
9.05	11.09	0	0	25.06	0
0	25.49	0	0	14.70	11.81
0	0	20.35	13.11	0	4.54
0	0	0	21.00	11.00	0

b) SMA

link loss flow , $\gamma = 432$, $\gamma = 3$

Table 10.8

on link (i,j) , but lost when finding the queue, conceptually connected to link (i,j) , filled. Table 10.8a corresponds to the CS policy and 10.8b to the SMA policy. It is notable that the number of lost packets is smaller under the SMA policy than under the CS policy for all links except link $(3,5)$. However, our cost function minimizes the total number of lost packets in every node and thus we can find effects like this.

10.6.1.3 Throughput

Figure 10.5 shows the total throughput as a function of gamma, i.e. the number of offered packets per second under load pattern i). Throughput is drawn under the SMA, the FSMA and the CS policies and the number of dedicated buffers under the FSMA policy is identical to that under the SMA policy, when $\gamma = 144$, i.e. $\gamma = 1$. For the presented interval we do not only find the SMA policy to be best, but in particular the FSMA policy to behave much better than the CS policy.

Figure 10.6 does also show total throughput as a function of gamma, but now for load pattern ii). Thus all GAMMA (s,t) except GAMMA $(2,6)$ are held constant and equal to those of table 10.5. Again we find the SMA policy superior to the CS policy and moreover total throughput under the CS and the FSMA policies form a decreasing function of offered load. We also find the CP policy doing very well under heavy load.

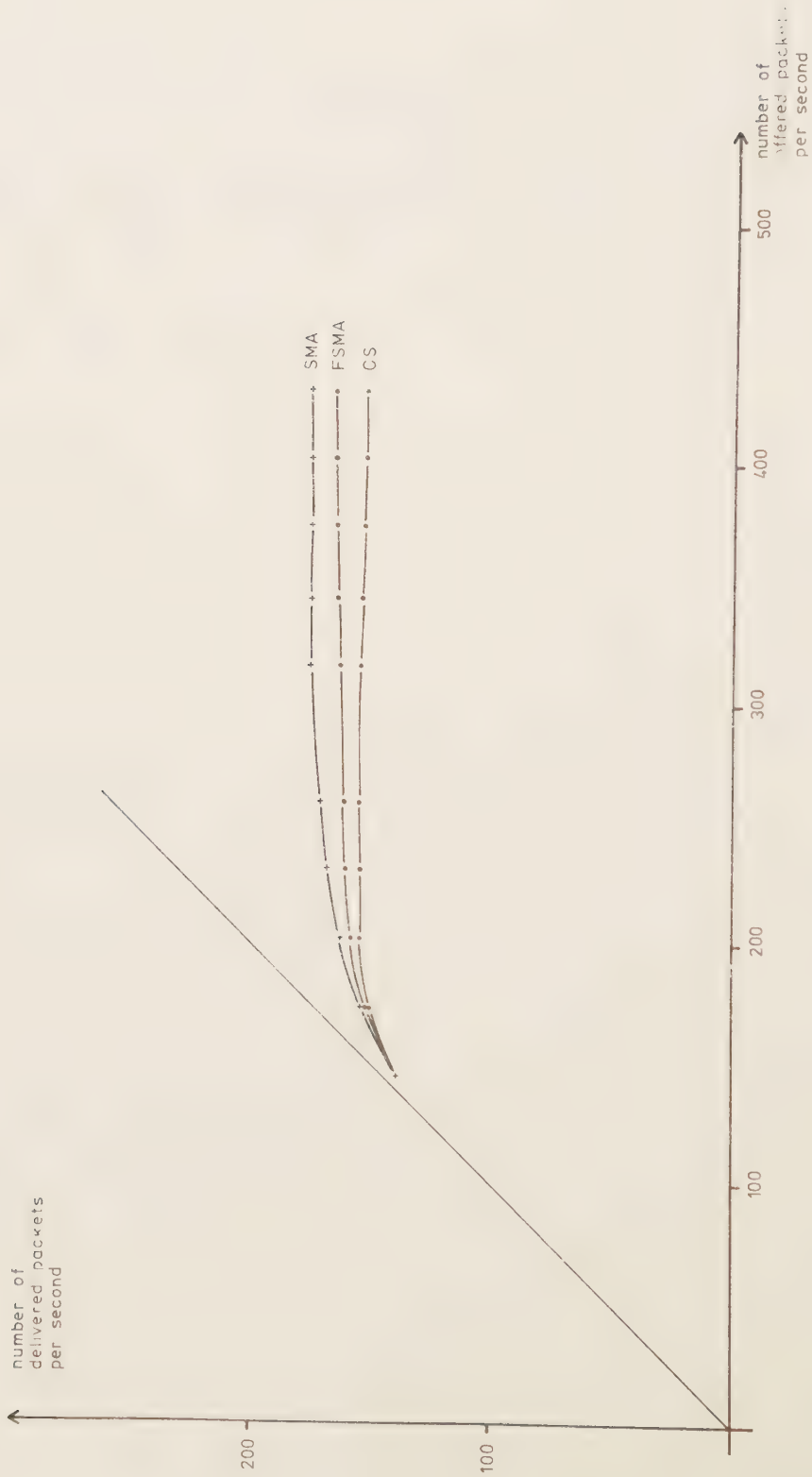


Figure 10.5

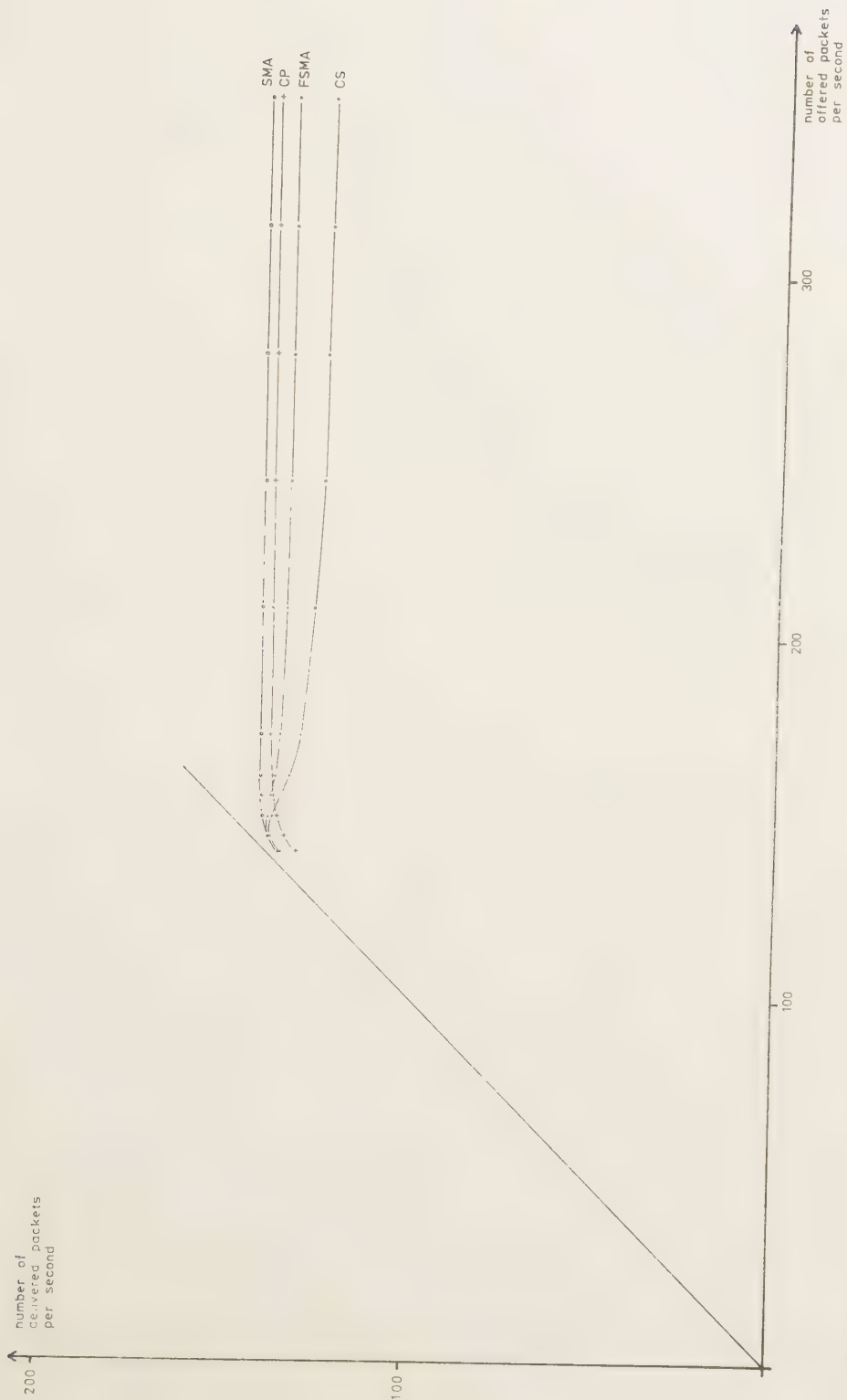


Figure 10.6

10.6.2 The 11-node net

For the 11-node net we let the arrival rates emanate from the following GAMMA matrix.

	0	6	6	6	20	6	1	4	4	1	1
	6	0	2	2	4	2	1	1	1	1	1
	6	2	0	2	4	2	1	1	1	1	1
	6	2	2	0	4	2	1	1	1	1	1
	20	4	4	4	0	4	1	2	2	1	1
GAMMA =	6	2	2	2	4	0	1	1	1	1	1
	1	1	1	1	1	1	0	4	4	2	2
	4	1	1	1	2	1	4	0	4	4	4
	4	1	1	1	2	1	4	4	0	4	4
	1	1	1	1	1	1	2	4	4	0	2
	1	1	1	1	1	1	2	4	4	2	0

Table 10.9

As for the 6-node net, the intensity matrix GAMMA is increased by a factor γ under load pattern i). Under load pattern ii) we present results for two different cases: The intensity from node 1 to 6, $\text{GAMMA}(1,6)$, and that from node 2 to 5, $\text{GAMMA}(2,5)$, are varied. Thus, when $\gamma = 1$ $\text{gamma} = \sum \sum \text{GAMMA} = 288$ packets per second. The number of dedicated buffers under the FSMA policy is equal to those of the SMA policy when $\text{gamma} = 460.8$, i.e. $\gamma = 1.6$, load pattern i).

For the 11-node net the total number of buffers in each node $M = 20$.

10.6.2.1 Offered load

Table 10.10 and 10.11 show offered load under the CS and

the SMA policies and for $\gamma = 1$ and $\gamma = 3$, load pattern i).

0	0.65	0	0	0	0.55	0.26	0	0	0	0
0.65	0	0.40	0	0	0	0	0	0	0	0
0	0.40	0	0.40	0	0	0	0	0	0	0
0	0	0.50	0	0.40	0.36	0	0	0	0	0
0	0	0	0.32	0	0.31	0	0	0.24	0	0
0.50	0	0	0.44	0.27	0	0	0	0	0	0
0.26	0	0	0	0	0	0	0.42	0	0	0.55
0	0	0	0	0	0	0.42	0	0.48	0	0.45
0	0	0	0	0.24	0	0	0.48	0	0.60	0
0	0	0	0	0	0	0	0	0.60	0	0.30
0	0	0	0	0	0	0.55	0.45	0	0.30	0

a) $\gamma = 1$

0	1.58	0	0	0	1.40	0.60	0	0	0	0
1.95	0	1.20	0	0	0	0	0	0	0	0
0	1.06	0	1.32	0	0	0	0	0	0	0
0	0	1.08	0	0.93	0.94	0	0	0	0	0
0	0	0	0.84	0	0.87	0	0	0.63	0	0
1.35	0	0	0.86	0.57	0	0	0	0	0	0
0.57	0	0	0	0	0	0	0.89	0	0	1.06
0	0	0	0	0	0	1.06	0	1.18	0	1.08
0	0	0	0	0.56	0	0	1.33	0	1.55	0
0	0	0	0	0	0	0	0	1.80	0	0.90
0	0	0	0	0	0	1.38	1.35	0	0.70	0

b) $\gamma = 3$

offered load, CS

Table 10.10

0	0.65	0	0	0	0.50	0.26	0	0	0	0
0.65	0	0.40	0	0	0	0	0	0	0	0
0	0.40	0	0.50	0	0	0	0	0	0	0
0	0	0.50	0	0.40	0.36	0	0	0	0	0
0	0	0	0.32	0	0.31	0	0	0.24	0	0
0.50	0	0	0.44	0.27	0	0	0	0	0	0
0.26	0	0	0	0	0	0	0.42	0	0	0.55
0	0	0	0	0	0	0.42	0	0.48	0	0.45
0	0	0	0	0.24	0	0	0.48	0	0.60	0
0	0	0	0	0	0	0	0	0.60	0	0.30
0	0	0	0	0	0	0.55	0.45	0	0.30	0

a) $\gamma = 1$

0	1.65	0	0	0	1.42	0.59	0	0	0	0
1.95	0	1.20	0	0	0	0	0	0	0	0
0	0.98	0	1.44	0	0	0	0	0	0	0
0	0	1.21	0	0.95	0.91	0	0	0	0	0
0	0	0	0.90	0	0.89	0	0	0.66	0	0
1.36	0	0	0.94	0.62	0	0	0	0	0	0
0.60	0	0	0	0	0	0	1.12	0	0	1.32
0	0	0	0	0	0	1.12	0	1.25	0	1.15
0	0	0	0	0.56	0	0	1.33	0	1.58	0
0	0	0	0	0	0	0	0	1.80	0	0.90
0	0	0	0	0	0	1.63	1.35	0	0.68	0

b) $\gamma = 3$

offered load, SMA

Table 10.11

Again tables 10.10a and 10.11a are identical, but 10.10b and 10.11b are not. Though total utilization under the SMA policy is larger than that under the CS policy, offered load is for some links larger under the CS policy than under the SMA policy. Take for instance node 1 where $\rho_{12} = 1.65$, $\rho_{16} = 1.42$ and $\rho_{17} = 0.59$ under SMA and $\rho_{12} = 1.58$, $\rho_{16} = 1.40$ and $\rho_{17} = 0.60$ under CS. On the contrary $\rho_{61} = 1.36$, $\rho_{64} = 0.94$ and $\rho_{65} = 0.62$ under SMA and $\rho_{61} = 1.35$, $\rho_{64} = 0.86$ and $\rho_{65} = 0.57$ under CS.

10.6.2.2 The number of lost packets

Table 10.12a and b shows the number of lost packets per second on link (i,j) under the CS and the SMA policies respectively. Comparing the two tables we find the number of lost packets dramatically reduced on some of the links, when using the SMA policy instead of the CS policy. In order to understand this reduction we have to look at the number of dedicated buffers in the SMA case. An element i,j in table 10.13 corresponds to the number of dedicated buffers allocated for j -packets in node i . As seen in table 10.12a and b, the number of lost packets on link $(1,7)$ is equal to 22.43 per second in the CS case and 0.75 in the SMA case (in node 1 we call them 7-packets), whereas the number of lost 2-packets is equal to 11.89 in the CS case and 13.79 in the SMA case. In table 10.13 we find that the SMA policy has allocated 6 buffers for 7-packets, but just 3 buffers for 2-packets. Having in mind that the SMA policy favours small loads and large arrival rates, the result is understandable as offered load is 0.60 and 1.57 for 7-packets and 2-packets respectively and the corresponding arrival rates are 59.30 and 33.08 packets per second respectively.

0	11.89	0	0	0	52.71	22.43	0	0	0	0
19.00	0	11.69	0	0	0	0	0	0	0	0
0	5.34	0	16.61	0	0	0	0	0	0	0
0	0	7.19	0	6.20	6.29	0	0	0	0	0
0	0	0	1.23	0	2.55	0	0	0.92	0	0
35.29	0	0	11.19	14.91	0	0	0	0	0	0
5.55	0	0	0	0	0	0	4.30	0	0	2.05
0	0	0	0	0	0	10.90	0	12.04	0	4.40
0	0	0	0	10.17	0	0	24.05	0	11.23	0
0	0	0	0	0	0	0	0	16.00	0	8.00
0	0	0	0	0	0	8.61	8.40	0	4.35	0

a) CS

0	13.79	0	0	0	42.75	0.75	0	0	0	0
19.11	0	4.29	0	0	0	0	0	0	0	0
0	1.52	0	22.14	0	0	0	0	0	0	0
0	0	12.86	0	4.43	3.72	0	0	0	0	0
0	0	0	2.40	0	3.27	0	0	0.92	0	0
37.86	0	0	4.14	1.18	0	0	0	0	0	0
1.07	0	0	0	0	0	0	8.55	0	0	7.47
0	0	0	0	0	0	8.93	0	14.23	0	5.31
0	0	0	0	0.45	0	0	17.26	0	12.00	0
0	0	0	0	0	0	0	0	16.16	0	0.52
0	0	0	0	0	0	18.00	7.49	0	0.34	0

b) SMA

link loss flow , gamma = 864 , $\gamma = 3$

Table 10.12

$\begin{array}{c c} & j \\ \hline i & \end{array}$	1	2	3	4	5	6	7	8	9	10	11
1	0	3	0	0	0	8	6	0	0	0	0
2	5	0	13	0	0	0	0	0	0	0	0
3	0	9	0	6	0	0	0	0	0	0	0
4	0	0	3	0	5	5	0	0	0	0	0
5	0	0	0	2	0	5	0	0	2	0	0
6	4	0	0	6	5	0	0	0	0	0	0
7	5	0	0	0	0	0	0	8	0	0	3
8	0	0	0	0	0	0	7	0	6	0	3
9	0	0	0	0	5	0	0	8	0	4	0
10	0	0	0	0	0	0	0	0	3	0	13
11	0	0	0	0	0	0	5	7	0	6	0

the number of dedicated buffers

Table 10.13

10.6.2.3 Throughput

Figure 10.7 shows throughput as a function of gamma under the CS and the SMA policies and under load pattern i). As seen in the diagram, the number of delivered packets per second form, under the SMA policy, a non-decreasing function of offered load, whereas that corresponding to the CS policy decreases. This last effect could be seen better in fig 10.8, where besides throughput for the CS and the SMA policies also those of the ESMA and the FSMA policies are drawn. Even though the FSMA policy in this case forms a non-decreasing function of offered load, this is not so in general. In figure 10.8 we can see very clearly, that the curve corresponding to the CS policy decreases as load increases.

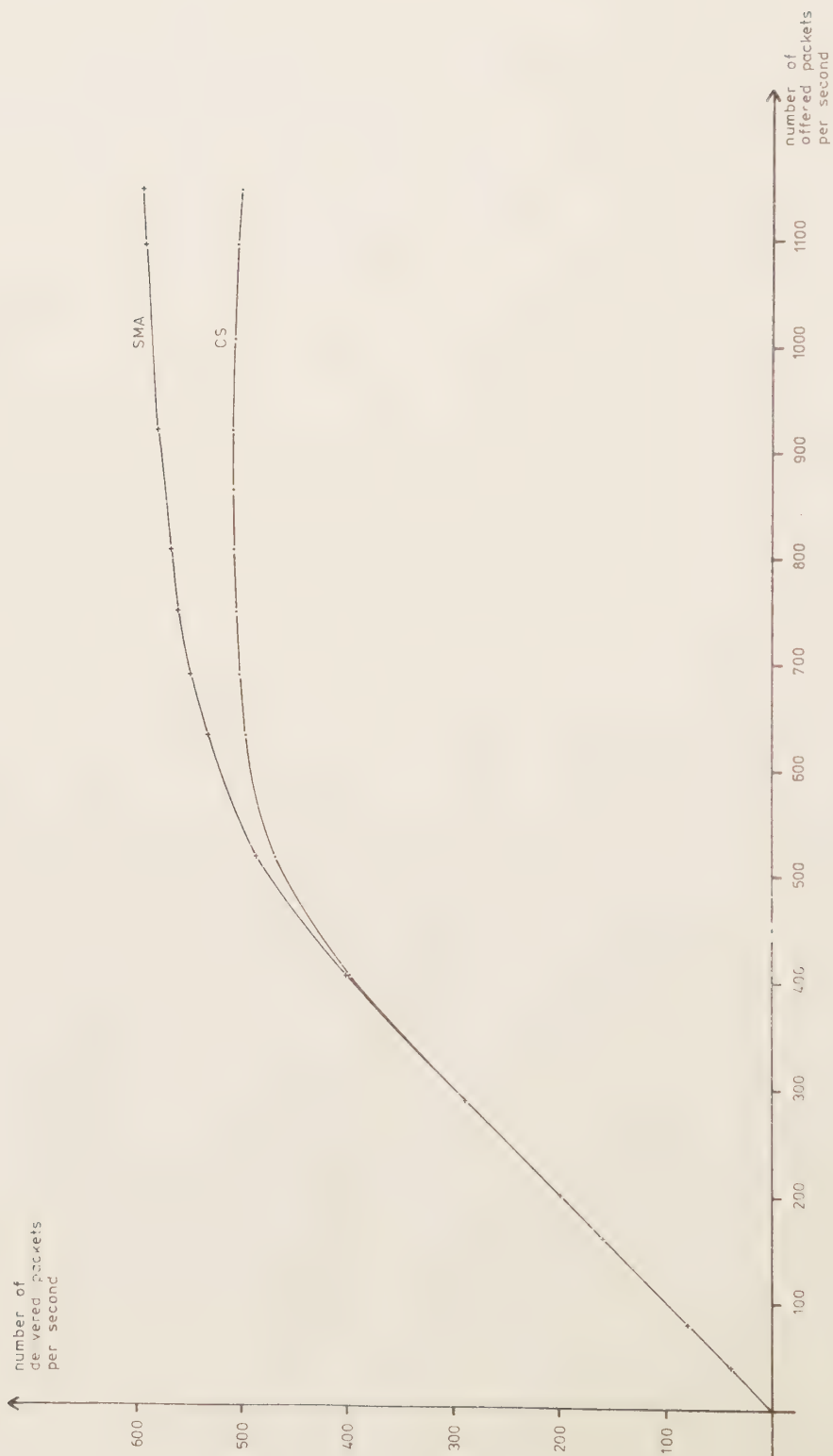


Figure 10.7

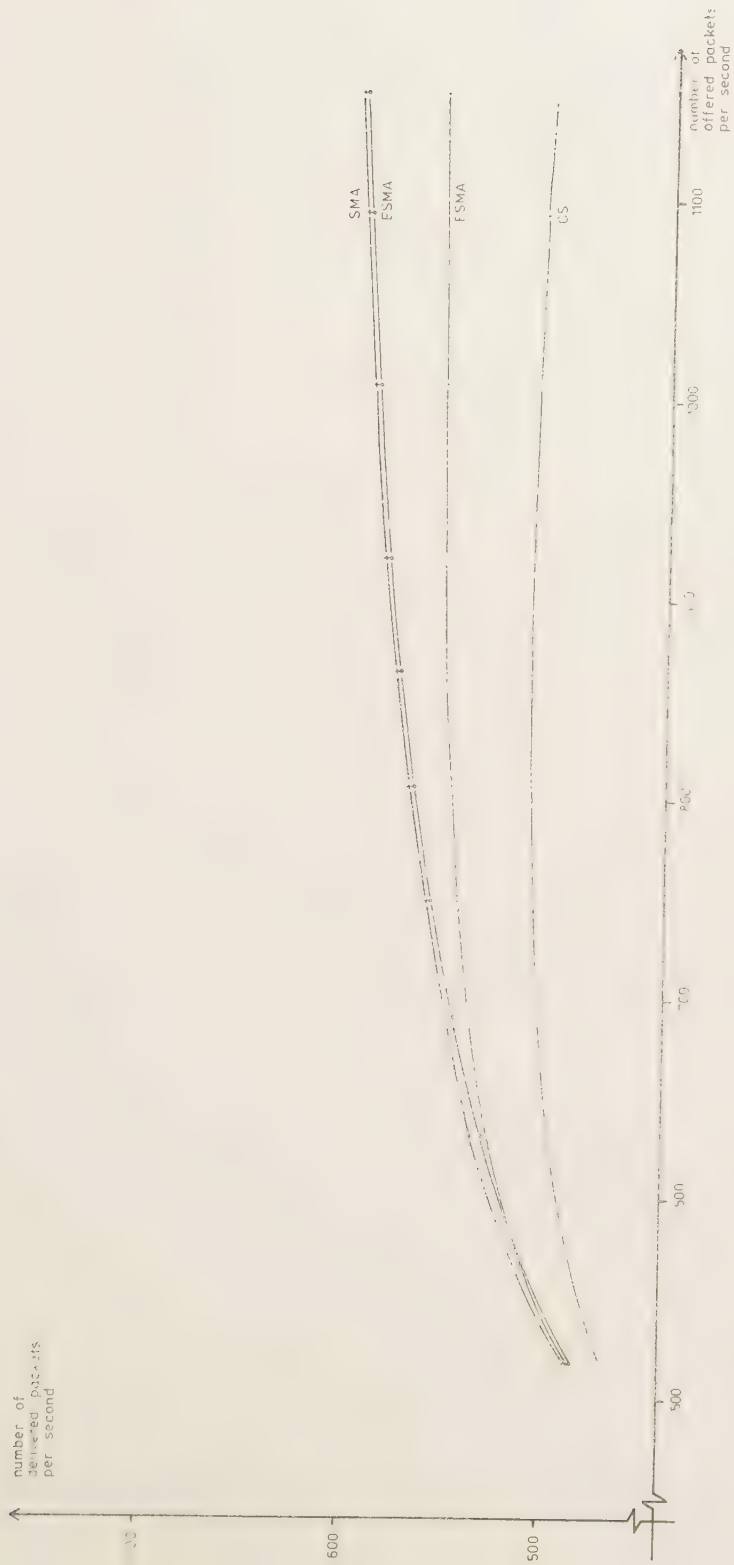


Figure 10.8

Figure 10.9 is based on load pattern ii), where all $\text{GAMMA}(s,t)$ except $\text{GAMMA}(1,6)$ are held constant. On link 1,6 offered load (or the part of offered load corresponding to packets with origin node 1 and destination node 6) is increased from 0.06 to 2.1. In figure 10.10 we present the interesting part of figure 10.9. Here we find the SMA as well as the CP policy behaving very well when $\text{GAMMA}(1,6)$ has increased dramatically. Furthermore we find the SMA policy forming a non-decreasing function of offered load.

In figures 10.9 and 10.10 we increased the packet rate between two adjacent nodes. If instead, under load pattern ii), we increase the source-destination packet rate between the two non-adjacent nodes 2 and 5, we will find that even throughput corresponding to the SMA policy will decrease, though to a small extent, with offered load.

In table 10.14 we have listed throughput for the different restricted buffer sharing policies as a function of gamma and we find that throughput decreases for all policies as load increases. Take for instance the FCP policy. How is it possible that throughput decreases when flow increases?

In order to realise that this effect is possible let us stress an example, where we more easily can understand the underlying reasons.

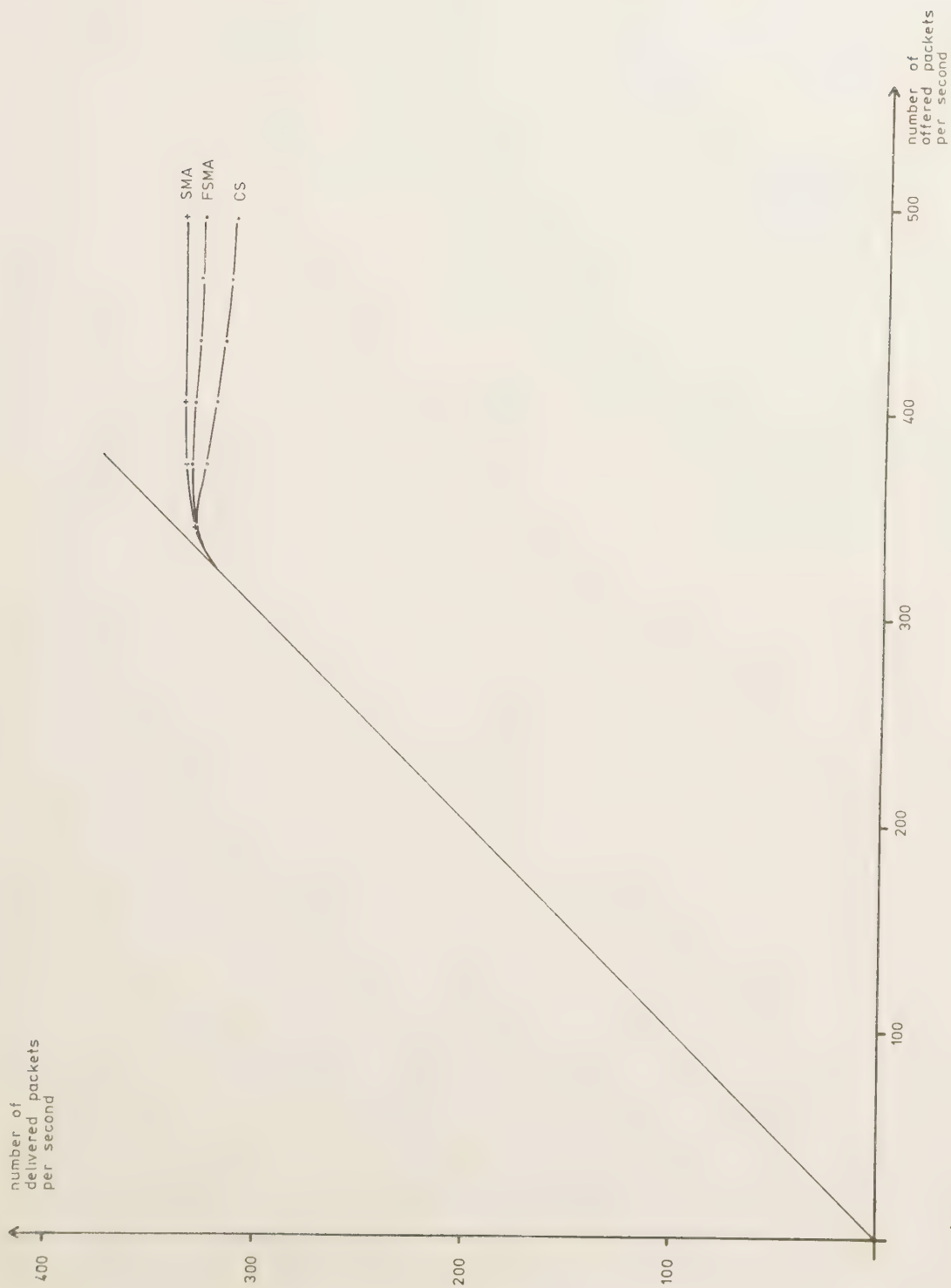


Figure 10.0

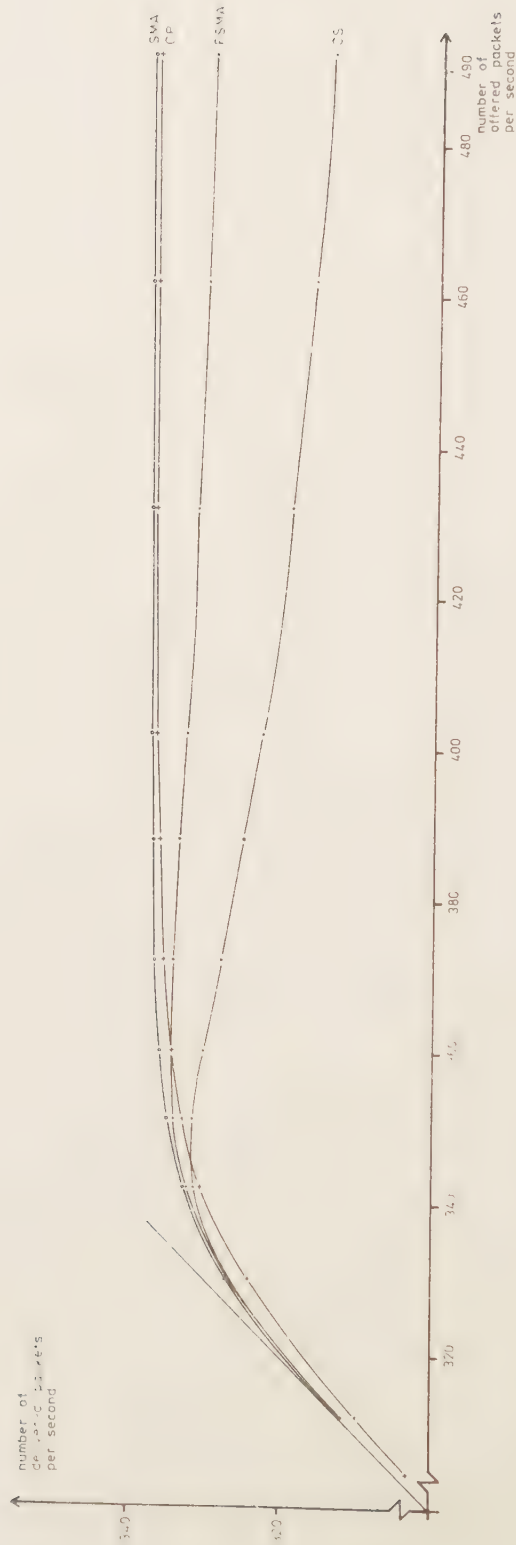


Figure 10.10

<u>Gamma</u>	<u>Gamma (2,5)</u>	<u>factor</u>	<u>CS</u>	<u>ESMA</u>	<u>FSMA</u>	<u>SMA</u>	<u>CP</u>	<u>FOP</u>
288	5	1.0	287.98	287.98	287.98	287.98	286.89	286.89
289.6	5.6	1.4	289.58	289.58	289.58	289.58	288.44	288.43
291.2	7.2	1.8	291.18	291.18	291.18	291.18	289.98	289.92
292.8	8.8	2.2	292.77	292.77	292.77	292.77	291.49	291.33
294.4	10.4	2.6	294.34	294.35	294.35	294.35	292.96	292.92
296	12.0	3.0	295.85	295.86	295.86	295.86	294.33	293.74
297.6	13.6	3.4	297.15	297.20	297.19	297.20	295.53	294.88
299.2	15.2	3.8	298.01	298.24	298.14	298.24	296.49	295.14
304	20	5	297.58	299.51	298.49	299.51	297.84	296.76
324	40	10	292.80	299.83	296.49	299.85	298.18	297.12
344	60	15	290.94	299.81	295.86	299.86	298.14	297.12
364	80	20	289.96	299.80	295.56	299.86	298.02	297.15
384	100	25	289.36	299.80	295.39	299.85	298.10	297.21
404	120	30	288.95	299.79	295.27	299.85	298.09	297.18
424	140	35	288.66	299.79	299.19	299.85	298.08	297.16

Table 10.14

Let us in a hypothetical net, see figure 10.11, observe the flow between node 1 and 5 and the flow between node 2 and 6. Let $C_{13} = C_{46} = 50$ kb/s, $C_{23} = C_{34} = 20$ kb/s and

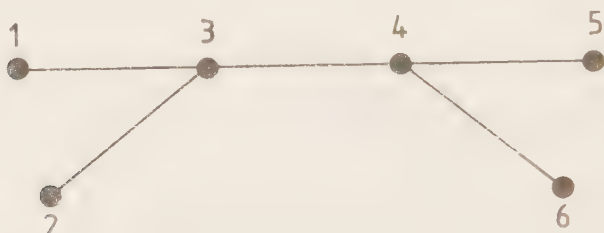


Figure 10.11

$C_{45} = 10$ kb/s. We assume that there are 10 buffers in each node and the FCP policy has given that in node 4 5-packets and 6-packets have 5 dedicated buffers each. In the other nodes there exists just one type of packets. Furthermore let $\lambda_{26} = 5$ packets per second and the mean packets length $1/\mu = 1000$ bits. Now we increase λ_{15} . When λ_{15} is equal to say 20 packets per second, link 3,4 is rather saturated. The probability of finding it non busy is approximately equal to $2.3 \cdot 10^{-2}$ and the departure rate is approximately equal to 19.5 packets per second. Among the packets leaving node 4, approximately $5/25$ are bound for node 6. This gives us an arrival rate of 15.6 packets per second on link 4,5 and 3.9 packets per second on link 4,6 and the total net throughput is approximately equal to $3.9 \cdot (1 - \text{Blocking probability on link 4,6}) + 15.6(1 - \text{Blocking probability on link 4,5})$. As link 4,5 is almost saturated and link 4,6 barely loaded, the total net throughput is approximately equal to $3.9 + 9.6 = 13.5$ packets per second.

If we now increase λ_{15} to say 45 packets per second, the departure rate from node 4 will not increase signi-

ificantly, but the proportion of packet types will change in favour of packets bound for node 5. Thus the arrival rate on link 4,5 is approximately $20 \cdot 45/50 = 18$ packets per second and that on link 4,6 is $20 \cdot 5/50 = 2$ packets per second and the net throughput is approximately equal to $2 + 9.8 = 11.8$ packets per second.

Even if this example contains odd link capacities, it shows that the discussed effects are possible.

10.6.2.4 Blocking probabilities

In section 10.5 we defined the mean path blocking probability as

$$\text{meanpathloss} = \frac{1}{\text{gamma}} \cdot \sum_s \sum_t \text{GAMMA}(s,t) \cdot \text{PATHLOSS}(s,t)$$

is the mean end to end blocking probability for an arbitrary packet. In figure 10.12 we have drawn $10 \log(\text{meanpathloss})$ as a function of gamma under load pattern i). We find the CP policy behaving poorly when gamma is small and to approach the curve corresponding to the SMA policy as gamma increases. We also find the CS policy doing very well under a light load, but poorly under heavy load.

In figure 10.13 we can better study the difference in a heavy load situation.

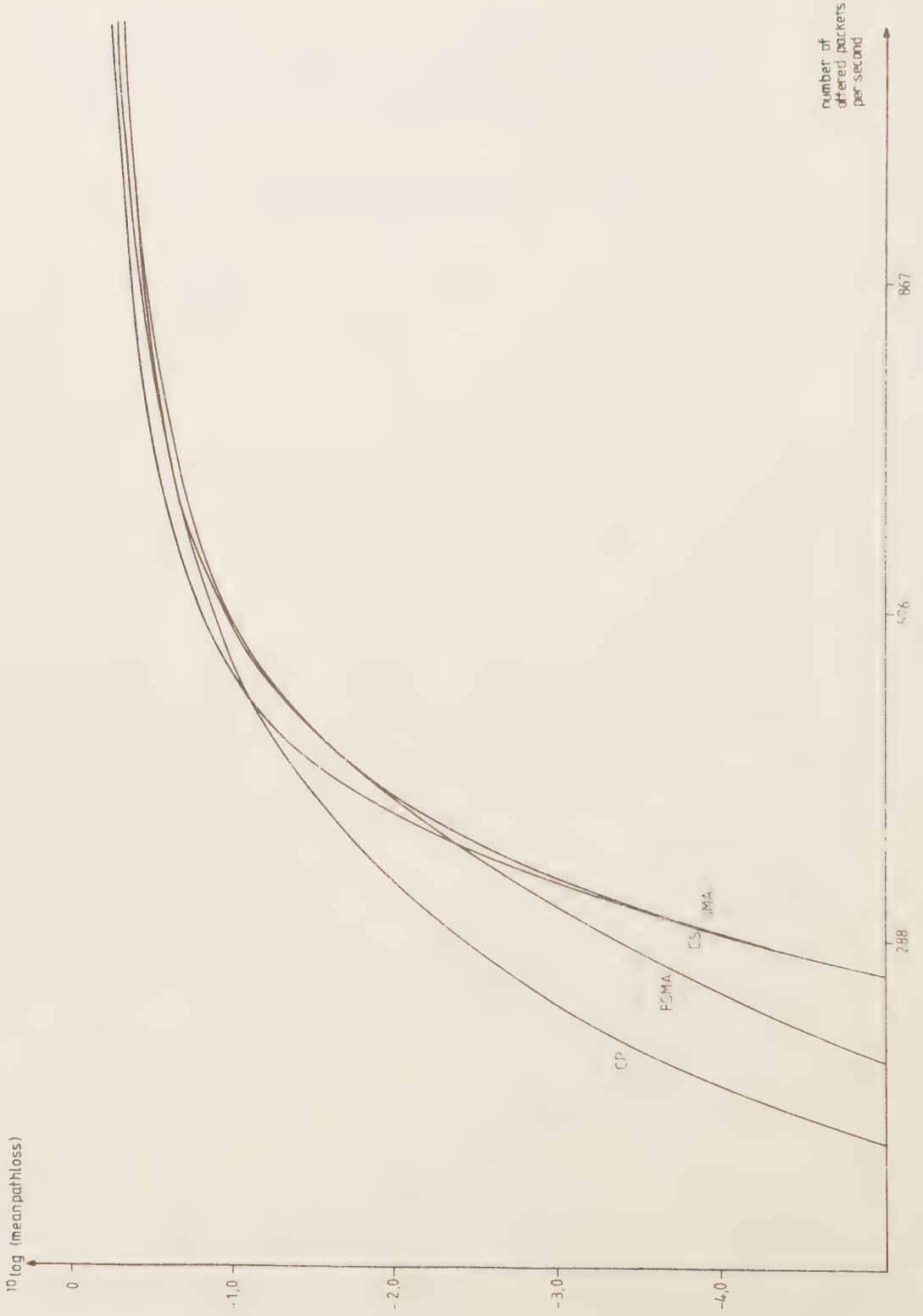


Figure 10.12

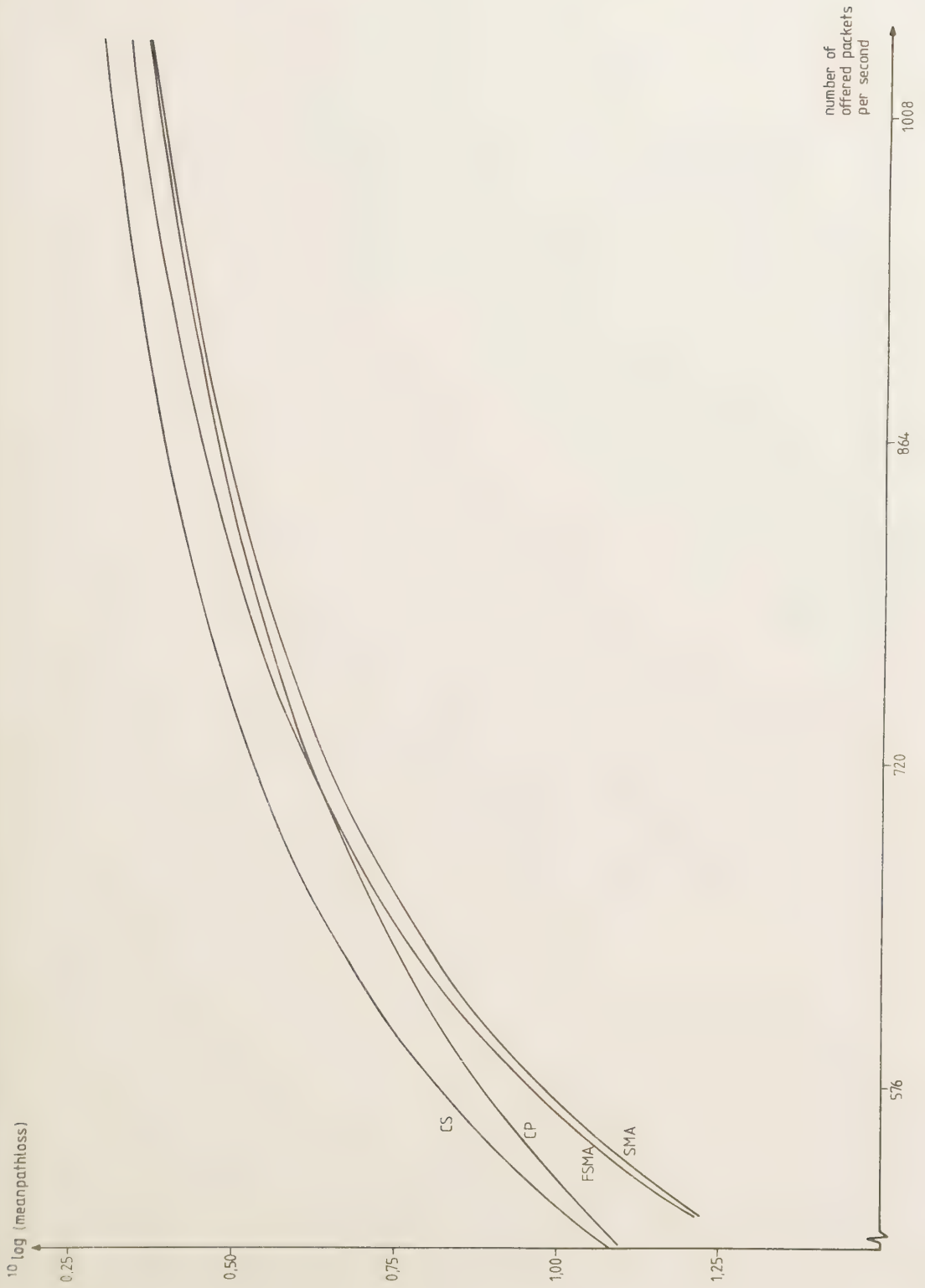


Figure 10.13

10.6.2.5 Delays

In chapter 6, when we presented delays in a node, i.e. the time a packet spent in one of the buffers, we found that delays as a function of load had leaps, when the number of dedicated buffers changed under the SMA policy. Those leaps were also found when using the CP policy. When we apply these policies to a complete net, the leaps in the curve corresponding to mean end to end delays appear when the number of dedicated buffers is changed in any node. As we increase the end to end packet rates, as we do under load pattern i), the allocation of the number of dedicated buffers will frequently change in the nodes. Thus the curves corresponding to the SMA and the CP policies (and also the ESMA policy though not being presented here) will be very jagged. In figure 10.14 mean path delay is drawn as a function of γ under the CS, the FSMA, the FCP, the SMA and the CP policies. We find great differences between the curves, but one should of course be aware of the blocking probabilities corresponding to the different policies. As we discussed in chapter 6 one could present a hypothetic restricted buffer sharing policy, which throws out most of the packets and gives the remaining packets a very short transit delay.



Figure 10.14

10.6.2.6 The mean number of packets in the net

When comparing the SMA and the CS policies with respect to throughput we have found the former to yield a higher throughput. When we compare the two policies with respect to delays again the SMA policy will have the best performance, i.e. yield a shorter total delay. Though the SMA policy contributes to a higher throughput, a net using the CS policy will on average have more packets in its node buffers. Furthermore, if we put our attention to the mean number of packets in the net that will be successfully delivered, the CS policy will do better than the SMA policy. Thus, even if the SMA policy will have a better throughput than the CS policy, the latter will contribute more to the number of successfully delivered packets in the network than the former. However, these performance measures are as we saw in section 10.5.2 related via Little's result and their ratio is equal to mean transit delay. The mean number of packets in the network, both total and successfully delivered, are drawn in figure 10.15.

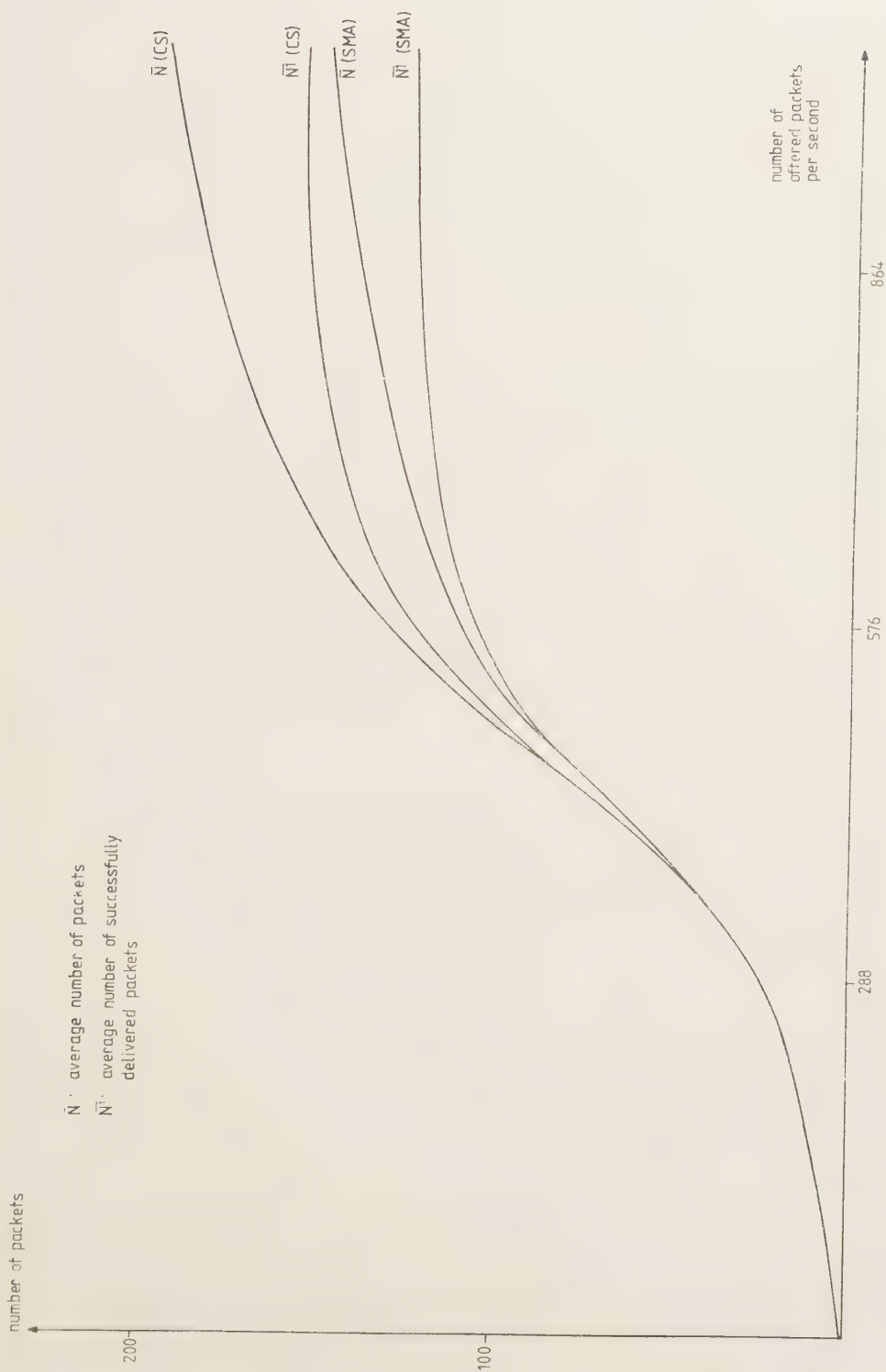


Figure 10.15

10.7 Conclusions

We have seen that what we achieved by introducing restricted buffer sharing policies in a packet switching node in chapter 6 regarding one single node was also achieved and reinforced in a complete network. The performance measures were found quite different in magnitude for the different policies.

However, an essential point in our work must be the credibility on different levels of our model. Here our general independence assumption is crucial. Simulations, in which Poisson streams to the different nodes are not assumed, only the source-destination rates are assumed Poisson, have shown good agreements with our analytic results. Such experiences have also been made in a number of other works [40],[53],[29]. Moreover measurements [52],[36] contribute to the credibility of using the general independence assumption in our model.

During the last five years much effort has been made in analysing packet switching computer communication networks under the assumption of finite storage capacities in the nodes. Most of these works assume a CS policy in the nodes.

Simon S Lam [59] early made an analysis of packet switching networks with finite buffer space in each node. He also incorporated the effects of positive acknowledgment signals in his model.

Peter H Meier [66],[67] uses queueing network theory applied on small queueing networks with finite storage capacities with general service distributions. However, he can only use the method on very small systems like

two queues in tandem.

F Borgonovo and L Fratta [8] analyse message swithing tandem and loop networks and take into account finite storage at nodes. They give upper and lower bounds to the node blocking probabilities when the exact values are not available because of the computational complexity.

J Labetoulle and G Pujolle [58] give a very fast, and consequently very economical, numerical solution, which gives the probabilities of saturation of each node, the mean response times of the network. They also give tools to determine whether a network can accept a given load without saturation or not. Their method is applied to the CYCLADES [73] network and the results are compared with simulations. Their model does incorporate a finite buffer assumption.

CHAPTER 11 - CONCLUSIONS AND FURTHER RESEARCH

11.1 Conclusions

This research was motivated by a need to understand how restricted buffer sharing policies can be used for congestion control in the nodes of a packet switching computer communication network. Though our models sometimes are rather simplistic, they can make us understand a basic network behaviour. We have found that the introduction of restricted buffer sharing policies improves the performance characteristics in terms of e g throughput and delay, and moreover protects the entire network from degradation under abnormal and heavy load situations.

The most important achievements can be summerized as follows: We have derived different models of a packet switching node with finite storage and restricted buffer sharing policies. The basic model, derived in chapter 6, was in chapter 8 extended to incorporate the effects of acknowledgement signals and retransmission. We introduced the concept "penalty functions" and incorporated these in the packet switching node model in order to give priority to packets that have reached their destination node and thus should not be lost. Finally, the packet switching node model was used in a global packet switching network model. We presented comprehensive results for the different models, as throughput, delays, blocking probabilities and buffer allocation schemes.

Furthermore, we derived a number of analytic tools introduced some new concepts and clarified some techniques used in our models to compute the main performance characteristics. The following is worthwhile to mention:

- The recursive technique to compute the normalization constant, the blocking probabilities and the mean number of packets in a node.
- The concept penalty functions.
- Algorithms to compute linkflows.
- Algorithms to compute end-to-end blocking probabilities and delays.
- A discussion on how to use matrix-algorithms to compute performance characteristics.
- A discussion on how to use Little's formula applied to a network with loss.

Throughout the chapters earlier analytic works on computer communication networks analyses were mentioned. It is many times very hard to make quantitative comparisons between results in one report with those in another, when the models are based on computer communication networks in general and not on a specific network. However, the results of other analyses and those of this research do often show good qualitative agreements. Furthermore, qualitative comparisons with simulations talk in favour to our basic assumptions like for instance the general independence assumption.

11.2 Further research

Analysis of computer communication networks requires development of more precise analytic network queueing models. In this research, the performance characteristics are presented in terms of average values. Variances and perhaps even higher moments must be derived in order to make better estimates of a network behaviour. In some cases, the assumption of fix routing in network models is quite inadequate. We must be able to incorporate

adaptive load dependent routing policies in the network models as well.

Up till now, the analyses of computer communication networks have been concentrated on small and medium sized networks. In the future we have to focus network analyses on larger networks with perhaps thousands of nodes and an enormous number of users. In these networks a large number of intelligent terminals may be linked directly to the network nodes and not via a host computer.

Further investigations have to be made regarding congestion control by means of restricted buffer sharing policies. Some of the policies, treated here, have showed valuable when used in small and medium sized networks. But new policies have to be developed for large networks and especially for large networks with many output links at each node (both towards other nodes and towards host computers or intelligent terminals). Furthermore, restricted buffer sharing policies combined with "penalty functions" have to be developed for the input queues at each node in order to throttle new traffic.

We have just started to analyse, one by one, the flow and congestion control mechanisms used in a net. There exists an urgent need to analyse the combined effects of such mechanisms.

APPENDIX A

A.1 Introduction

In our packet switching node models, treated in chapter 6 and chapter 8, we found the states $\underline{k}$, i.e. k_i i-packets ($\forall i = 1, 2, \dots, n$) in the buffers and $(\underline{k}, \underline{j})$, i.e. ($\forall i = 1, 2, \dots, n$) k_i i-packets waiting for transmission or for transmission to be completed and j_i i-packets waiting for an acknowledgement signal in the buffers respectively forming Markov chains. Though we are aware of that, among others, Lam [60] has shown that the steady state distribution for a Markov chain, similar to the one we gained in our model in chapter 6, could be seen to have a product form solution, we will in this appendix introduce a technique to derive the steady state probabilities corresponding to our models in both chapter 6 and chapter 8. The technique used is maybe a little too troublesome, however. We want to show a technique useful in more general problems. Moreover, in contradiction to our chain corresponding to our model in chapter 8, the one corresponding to our model in chapter 6 will form an n-dimensional birth-death process.

In section A.2, A.3 and A.4 we derive the steady state probability distributions corresponding to the models of a node without acknowledgement signals, a node with acknowledgement signals and a node with acknowledgement signals and retransmission respectively.

A.2 The steady state probability distribution of a limited two dimensional birth-death process

Our model of the packet switching node in chapter 6 leads to that the states $\underline{k}$, i.e. the state corresponding

k_1 1-packets, k_2 2-packets, ..., k_n n-packets in the buffer system, form an n-dimensional Markov chain, which also is a birth-death process. In this section we will, under the SMA policy, derive the steady state probability distribution using the global balance equations gained when claiming

$$\text{flow out from a state } \underline{k} = \text{flow into a state } \underline{k}$$

Furthermore, we restrict ourselves to the case when $n=2$ and we claim $C_1 = C_2 = C$. The technique we use to derive the steady state probabilities can be generalized to the case where $n > 2$ and $C_i \neq C_j \quad \forall i, j = 1, 2, \dots, n, i \neq j$.

In our birth-death process every state $\underline{k}$ is aperiodic, recurrent nonnull (see [51],[23]). This leads to that our two dimensional birth-death process is ergodic and thus there exists a unique equilibrium probability distribution with non-negative probabilities and this distribution is equal to the steady state probability distribution [23].

The two dimensional state space is given in figure A.1.

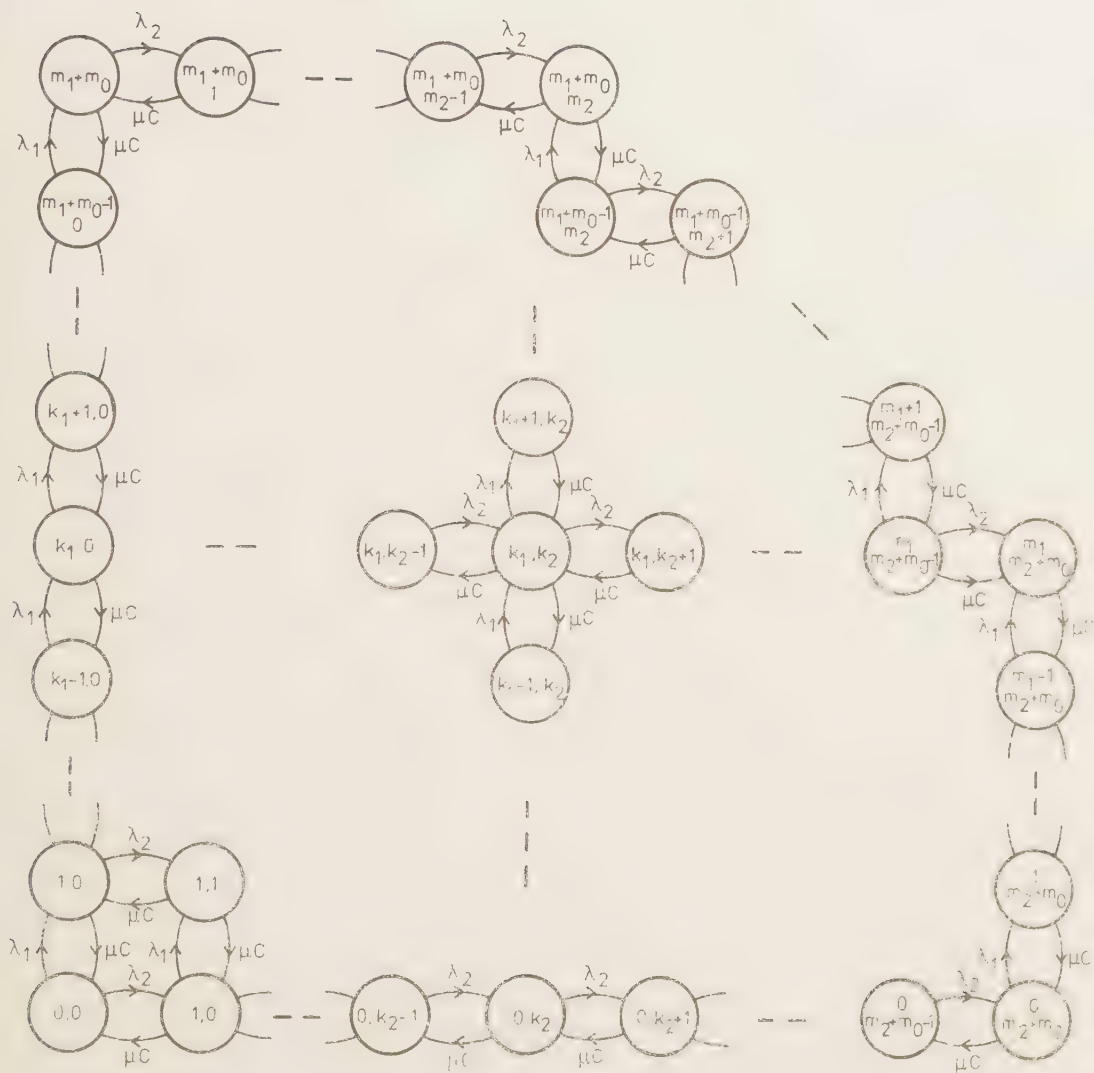


Figure A.1

A.2.1 The global balance equations

The global balance equations correspond to the areas indicated in figure A.2

$$\begin{aligned} \text{a) } (\lambda_1 + \lambda_2 + 2\mu C)P(k_1, k_2) &= \mu C(P(k_1, k_2 + 1) + P(k_1 + 1, k_2)) + \\ &+ \lambda_1 P(k_1 - 1, k_2) + \lambda_2 P(k_1, k_2 - 1) \end{aligned} \quad (\text{A.1})$$

$$\text{b) } (\lambda_1 + \lambda_2)P(0, 0) = \mu C(P(0, 1) + P(1, 0)) \quad (\text{A.2})$$

$$\begin{aligned} \text{c) } (\lambda_1 + \lambda_2 + \mu C)P(0, k_2) &= \mu C(P(0, k_2 + 1) + P(1, k_2)) + \lambda_2 P(0, k_2 - 1) \\ & \quad (\text{A.3}) \end{aligned}$$

$$\begin{aligned} \text{d) } (\lambda_1 + \lambda_2 + \mu C)P(k_1, 0) &= \mu C(P(k_1 + 1, 0) + P(k_1, 1)) + \lambda_1 P(k_1 - 1, 0) \\ & \quad (\text{A.4}) \end{aligned}$$

$$\begin{aligned} \text{e) } (\mu C + \lambda_2)P(m_0 + m_1, 0) &= \lambda_1 P(m_0 + m_1 - 1, 0) + \mu C P(m_0 + m_1, 1) \\ & \quad (\text{A.5}) \end{aligned}$$

$$\begin{aligned} \text{f) } (\mu C + \lambda_1)P(0, m_0 + m_2) &= \lambda_2 P(0, m_0 + m_2 - 1) + \mu C P(1, m_0 + m_2) \\ & \quad (\text{A.6}) \end{aligned}$$

$$\begin{aligned} \text{g) } (2\mu C + \lambda_2)P(m_0 + m_1, k) &= \lambda_1 P(m_0 + m_1 - 1, k) + \lambda_2 P(m_0 + m_1, k - 1) + \\ &+ \mu C P(m_0 + m_1, k + 1) \end{aligned} \quad (\text{A.7})$$

$$\begin{aligned} \text{i) } (2\mu C + \lambda_1)P(k, m_0 + m_2) &= \lambda_1 P(k - 1, m_0 + m_2) + \lambda_2 P(k, m_0 + m_2 - 1) + \\ &+ \mu C P(k + 1, m_0 + m_2) \end{aligned} \quad (\text{A.8})$$

A.2.2 The z-transform

In order to obtain the steady state distribution from the global balance equations we define the z-transform

$$F(z_1, z_2) = \sum_{\Omega} P(k_1, k_2) z_1^{k_1} z_2^{k_2} \quad (A.10)$$

where Ω is the limited area of our birth-death process.

$$\begin{aligned} F(z_1, z_2) = & \sum_{k_1=0}^{m_1} \sum_{k_2=0}^{m_2+m_0} P(k_1, k_2) z_1^{k_1} z_2^{k_2} + \\ & + \sum_{k_1=m_1+1}^{m_1+m_0} \sum_{k_2=0}^{M-k_1} P(k_1, k_2) z_1^{k_1} z_2^{k_2}; \end{aligned} \quad (A.11)$$

From this definition we find that

$$a) F(z_1, 0) = \sum_{k_1=0}^{m_1+m_0} P(k_1, 0) z_1^{k_1} \quad (A.12)$$

$$b) F(0, z_2) = \sum_{k_2=0}^{m_2+m_0} P(0, k_2) z_2^{k_2} \quad (A.13)$$

$$c) \frac{d^{m_2+m_0} F(z_1, z_2)}{dz_2^{m_2+m_0}} = (m_2+m_0)! \sum_{k_1=0}^{m_1} P(k_1, m_2+m_0) z_1^{k_1} \quad (A.14)$$

$$d) \frac{d^{m_1+m_0} F(z_1, z_2)}{dz_1^{m_1+m_0}} = (m_1+m_0)! \sum_{k_2=0}^{m_2+m_0} P(m_1+m_0, k_2) z_2^{k_2} \quad (A.15)$$

The z-transform a - d could be seen as the z-transform of our process over the areas indicated in figure A.3.

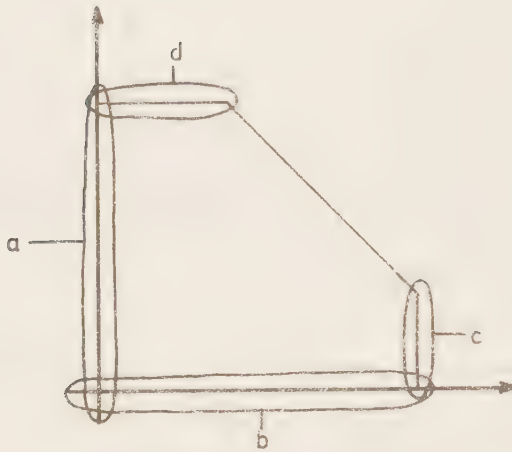


Figure A.3

For short let

$$F_{m_1+m_0} = \frac{1}{(m_1+m_0)!} \frac{d^{m_1+m_0} F(z_1, z_2)}{dz_1^{m_1+m_0}} z_1^{m_1+m_0} \quad (\text{A.16})$$

$$F_{m_2+m_0} = \frac{1}{(m_2+m_0)!} \frac{d^{m_2+m_0} F(z_1, z_2)}{dz_2^{m_2+m_0}} z_2^{m_2+m_0} \quad (\text{A.17})$$

and

$$F_M = \sum_{k=m_1}^{m_1+m_0} P(k, M-k) z_1^k z_2^{M-k} \quad (\text{A.18})$$

Multiplying the global equation A.1 by $z_1^{k_1} z_2^{k_2}$ and forming the sum of all such multiplied equations we get

$$\begin{aligned}
 \text{a) } (\lambda_1 + \lambda_2 + 2\mu C) \sum_{\Omega_1} P(k_1, k_2) z_1^{k_1} z_2^{k_2} &= \frac{\mu C}{z_2} \sum_{\Omega_2} P(k_1, k_2) z_1^{k_1} z_2^{k_2} + \\
 &+ \frac{\mu C}{z_1} \sum_{\Omega_3} P(k_1, k_2) z_1^{k_1} z_2^{k_2} + \lambda_1 z_1 \sum_{\Omega_4} P(k_1, k_2) z_1^{k_1} z_2^{k_2} + \\
 &+ \lambda_2 z_2 \sum_{\Omega_5} P(k_1, k_2) z_1^{k_1} z_2^{k_2} ; \quad (\text{A.19})
 \end{aligned}$$

The different Ω_i 's can be found in figure A.4.

The same technique applied to equations A.2-A.9 gives us

$$\text{b) } (\lambda_1 + \lambda_2) P(0, 0) = \frac{\mu C}{z_2} P(0, 1) z_2 + \frac{\mu C}{z_1} P(1, 0) z_1 \quad (\text{A.20})$$

$$\begin{aligned}
 \text{c) } (\lambda_1 + \lambda_2 + \mu C) \sum_{\Omega_1} P(0, k_2) z_2^{k_2} &= \frac{\mu C}{z_2} \sum_{\Omega_{10}} P(0, k_2) z_2^{k_2} + \\
 &+ \frac{\mu C}{z_1} \sum_{\Omega_{11}} P(1, k_2) z_1 z_2^{k_2} + \lambda_2 z_2 \sum_{\Omega_{12}} P(0, k_2) z_2^{k_2} \quad (\text{A.21})
 \end{aligned}$$

$$\begin{aligned}
 \text{d) } (\lambda_1 + \lambda_2 + \mu C) \sum_{\Omega_{13}} P(k, 0) z_1^{k_1} &= \frac{\mu C}{z_1} \sum_{\Omega_{14}} P(k_1, 0) z_1^{k_1} + \\
 &+ \frac{\mu C}{z_2} \sum_{\Omega_{15}} P(k_1, 1) z_1^{k_1} z_2 + \lambda_1 z_1 \sum_{\Omega_{16}} P(k_1, 0) z_1^{k_1} \quad (\text{A.22})
 \end{aligned}$$

$$\begin{aligned}
 \text{e) } (\mu + \lambda_2) P(m_0 + m_1, 0) z_1^{m_0 + m_1} &= \lambda_1 z_1 P(m_0 + m_1 - 1, 0) z_1^{m_0 + m_1 - 1} + \\
 &+ \frac{\mu C}{z_2} P(m_0 + m_1, 1) z_1^{m_0 + m_1} z_2 \quad (\text{A.23})
 \end{aligned}$$

$$\begin{aligned}
 f) \quad & (\mu C + \lambda_1) P(0, m_0 + m_2) z_2^{m_0 + m_2} = \lambda_2 z_2 P(0, m_0 + m_2 - 1) z_2^{m_0 + m_2 - 1} + \\
 & \Omega_{31} \qquad \qquad \qquad \Omega_{32} \\
 & + \frac{\mu C}{z_1} P(1, m_0 + m_2) z_1 \cdot z_2^{m_0 + m_2} \qquad \qquad \qquad (A.24) \\
 & \Omega_{33}
 \end{aligned}$$

$$\begin{aligned}
 g) \quad & (2\mu C + \lambda_2) \sum_{\Omega_{17}} P(m_0 + m_1, k) z_1^{m_0 + m_1} z_2^k = \\
 & = \lambda_1 z_1 \sum_{\Omega_{18}} P(m_0 + m_1 - 1, k) z_1^{m_0 + m_1 - 1} \cdot z_2^k + \\
 & + \lambda_2 z_2 \sum_{\Omega_{19}} P(m_0 + m_1, k) z_1^{m_0 + m_1} \cdot z_2^k + \\
 & + \frac{\mu C}{z_2} \sum_{\Omega_{20}} P(m_0 + m_1, k) z_1^{m_0 + m_1} z_2^k \qquad \qquad \qquad (A.25)
 \end{aligned}$$

$$\begin{aligned}
 i) \quad & (2\mu C + \lambda_1) \sum_{\Omega_{21}} P(k, m_0 + m_2) z_1^k z_2^{m_0 + m_2} = \\
 & = \lambda_1 z_1 \sum_{\Omega_{22}} P(k, m_0 + m_2) z_1^k z_2^{m_0 + m_2} + \\
 & + \lambda_2 z_2 \sum_{\Omega_{23}} P(k, m_0 + m_2 - 1) z_1^k z_2^{m_0 + m_2 - 1} + \\
 & + \frac{\mu C}{z_1} \sum_{\Omega_{24}} P(k, m_0 + m_2) z_1^k z_2^{m_0 + m_2} \qquad \qquad \qquad (A.26)
 \end{aligned}$$

$$k) \quad 2\mu C \sum_{\Omega_{25}} P(m_1 + m_0 - k, m_2 + k) z_1^{m_1 + m_0 - k} z_2^{m_2 + k} =$$

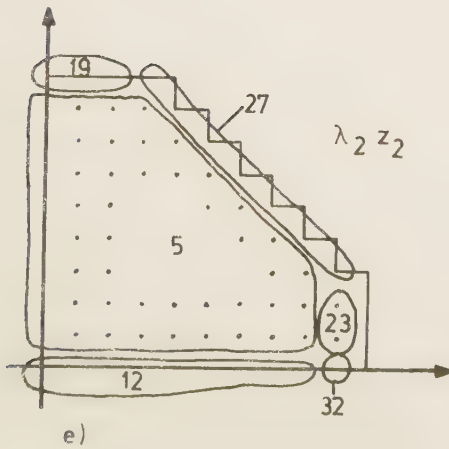
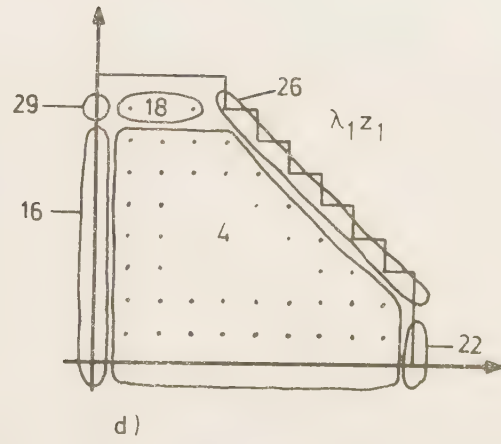
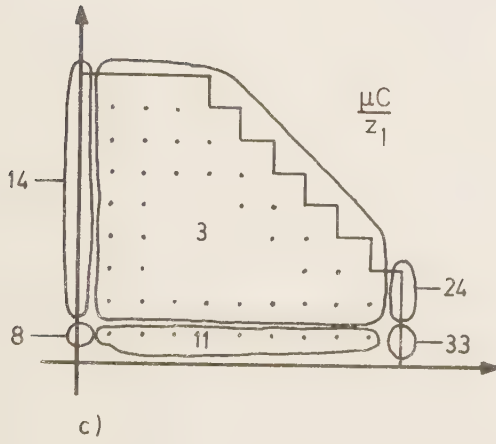
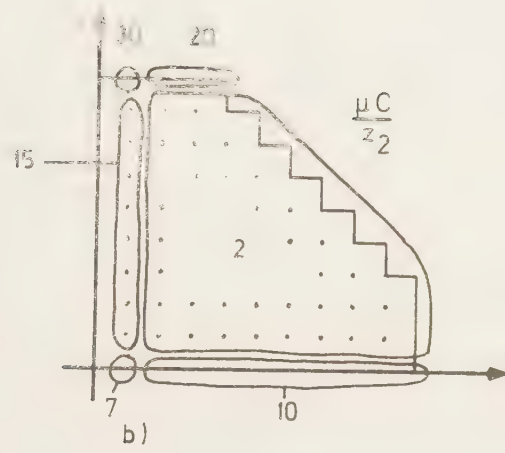
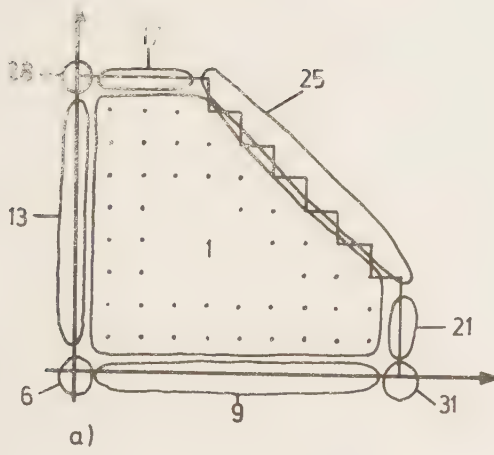


Figure A.4

$$\begin{aligned}
 &= \lambda_1 z_1 \sum_{\Omega_{26}} P(m_1+m_0-k-1, m_2+k) z_1^{m_1+m_0-k-1} z_2^{m_2+k} + \\
 &+ \lambda_2 z_2 \sum_{\Omega_{27}} P(m_1+m_0-k, m_2+k-1) z_1^{m_1+m_0-k} z_2^{m_2+k-1} \quad (A.27)
 \end{aligned}$$

In the following we refer to the different terms in the modified global balance equations as the number of the given state space Ω_i .

Let us form the sum of equations A.19 to A.27. From figure A.4.a we see that the left part of the formed equation, the sum of the terms 1, 6, 9, 13, 17, 21, 25, 28 and 31, equals

$$\begin{aligned}
 &(\lambda_1 + \lambda_2 + 2\mu C) F(z_1, z_2) - \mu C \sum_{k_2=1}^{m_2+m_0-1} P(0, k_2) z_2^{k_2} - \\
 &- \mu C \sum_{k_1=1}^{m_1+m_0-1} P(k_1, 0) z_1^{k_1} - \lambda_1 \sum_{k_2=1}^{m_2-1} P(m_0+m_1, k_2) z_1^{m_0+m_1} z_2^{k_2} - \\
 &- \lambda_2 \sum_{k_1=1}^{m_1-1} P(k_1, m_2+m_0) z_1^{k_1} z_2^{m_2+m_0} - \\
 &- (\lambda_1 + \lambda_2) \sum_{k=0}^{m_3} P(m_1+m_0-k, m_2+k) z_1^{m_1+m_0-k} z_2^{m_2+k} - 2\mu C P(0, 0) - \\
 &- \lambda_1 P(m_1+m_0, 0) z_1^{m_1+m_0} - \lambda_2 P(0, m_2+m_0) z_2^{m_0+m_2} - \\
 &- \mu C P(m_1+m_0, 0) z_1^{m_1+m_0} - \mu C P(0, m_2+m_0) z_2^{m_2+m_0} \quad (A.28)
 \end{aligned}$$

This could be rewritten as

$$\begin{aligned}
 & (\lambda_1 + \lambda_2 + 2\mu C) F(z_1, z_2) - \mu C F(0, z_2) - \mu C F(z_1, 0) - \lambda_1 F_{m_1+m_0} - \\
 & - \lambda_2 F_{m_2+m_0} - (\lambda_1 + \lambda_2) F_M
 \end{aligned} \tag{A:29}$$

In figure A.4 we have indicated the Ω_i :s corresponding to terms preceded by $\frac{\mu C}{z_2}$ in A.4.b, by $\frac{\mu C}{z_1}$ in A.4.c, by $\lambda_1 z_1$ in A.4.d and by $\lambda_2 z_2$ in A.4.e. Taking the terms preceded by $\frac{\mu C}{z_2}$ in the right part of our equation, number 2, 7, 10, 15, 20 and 30, we get

$$\frac{\mu C}{z_2} (F(z_1, z_2) - F(z_1, 0)) \tag{A.30}$$

The terms preceded by $\frac{\mu C}{z_1}$: 3, 8, 11, 14, 24 and 33

$$\frac{\mu C}{z_1} (F(z_1, z_2) - F(0, z_2)) \tag{A.31}$$

The terms preceded by $\lambda_1 z_1$: 4, 16, 18, 22, 26 and 29

$$\lambda_1 z_1 (F(z_1, z_2) - F_M - F_{m_1+m_0} + P(m_1+m_0, m_2) z_1^{m_1+m_0} z_2^{m_2}) \tag{A.32}$$

and finally the terms preceded by $\lambda_2 z_2$: 5, 12, 19, 23, 27, 32

$$\lambda_2 z_2 (F(z_1, z_2) - F_M - F_{m_2+m_0} + P(m_1, m_2+m_0) z_1^{m_1} z_2^{m_2+m_0}) \tag{A.33}$$

Thus the whole equation can be written

$$\begin{aligned}
 & F(z_1, z_2) (\lambda_1 + \lambda_2 + 2\mu C - \frac{\mu C}{z_1} - \frac{\mu C}{z_2} - \lambda_1 z_1 - \lambda_2 z_2) - \\
 & - F(z_1, 0) (\mu C - \frac{\mu C}{z_2}) - F(0, z_2) (\mu C - \frac{\mu C}{z_1}) - F_M (\lambda_1 + \lambda_2 - \lambda_1 z_1 - \lambda_2 z_2) - \\
 & - F_{m_1+m_0} (\lambda_1 - \lambda_1 z_1) - F_{m_2+m_0} (\lambda_2 - \lambda_2 z_2) + \\
 & + P(m_1+m_0, m_2) z_1^{m_1+m_2} z_2^{m_2} (\lambda_1 - \lambda_1 z_1) + \\
 & + P(m_1, m_2+m_0) z_1^{m_1} z_2^{m_2+m_0} (\lambda_2 - \lambda_2 z_2) = 0 \quad (A.34)
 \end{aligned}$$

Let us rewrite the equation

$$\begin{aligned}
 & (\mu C - \frac{\mu C}{z_1}) (F(z_1, z_2) (1 - \frac{\lambda_1}{\mu C} z_1) - F(0, z_2) + \frac{\lambda_1}{\mu C} z_1 F_{m_1+m_0, M}) + \\
 & + (\mu C - \frac{\mu C}{z_2}) (F(z_1, z_2) (1 - \frac{\lambda_2}{\mu C} z_2) - F(z_1, 0) + \frac{\lambda_2}{\mu C} z_2 F_{m_2+m_0, M}) = 0 \quad (A.35)
 \end{aligned}$$

where

$$F_{m_1+m_0, M} = F_{m_1+m_0} + F_M - P(m_1+m_0, m_2) z_1^{m_1+m_0} z_2^{m_2}; \quad (A.36)$$

and

$$F_{m_2+m_0, M} = F_{m_2+m_0} + F_M - P(m_1, m_2+m_0) z_1^{m_1} z_2^{m_2+m_0}; \quad (A.37)$$

As our Markov chain is ergodic, there exists a unique steady state distribution. This implies that if

$$F(z_1, z_2) (1 - \frac{\lambda_1}{\mu C} z_1) - F(0, z_2) + \frac{\lambda_1}{\mu C} z_1 F_{m_1+m_0, M} = 0 \quad (A.38)$$

$$F(z_1, z_2) (1 - \frac{\lambda_2}{\mu C} z_2) - F(z_1, 0) + \frac{\lambda_2}{\mu C} z_2 F_{m_2+m_0, M} = 0 \quad (A.39)$$

give a unique solution, $F(z_1, z_2)$, then it is the solution to eq A.35.

Let $z_2 = 0$ in eq A.38)

$$(1 - \frac{\lambda_1}{\mu C} z_1) F(z_1, 0) - F(0, 0) + \frac{\lambda_1}{\mu C} z_1 P(m_1+m_0, 0) z_1^{m_1+m_0} = 0 \quad (A.40)$$

$\Rightarrow$

$$F(z_1, 0) = \frac{F(0, 0)}{1 - \frac{\lambda_1}{\mu C} z_1} - \frac{\frac{\lambda_1}{\mu C} z_1}{1 - \frac{\lambda_1}{\mu C} z_1} P(m_1+m_0, 0) z_1^{m_1+m_0} \quad (A.41)$$

And in the same way

$$F(0, z_2) = \frac{F(0, 0)}{1 - \frac{\lambda_2}{\mu C} z_2} - \frac{\frac{\lambda_2}{\mu C} z_2}{1 - \frac{\lambda_2}{\mu C} z_2} P(0, m_2+m_0) z_2^{m_2+m_0} \quad (A.42)$$

In order to express $P(m_1+m_0, 0)$ as a function of $F(0, 0)$ we write

$$F(z_1, 0) = (F(0, 0) - \frac{\lambda_1}{\mu C} z_1 P(m_1+m_0, 0) z_1^{m_1+m_0}) \sum_{k=0}^{\infty} (\frac{\lambda_1}{\mu C} z_1)^k \quad (A.43)$$

and as the sum of the coefficients before $z_1^{m_1+m_0+1}$ must be equal to zero we find

$$F(0, 0) (\frac{\lambda_1}{\mu C})^{m_1+m_0+1} - \frac{\lambda_1}{\mu C} P(m_1+m_0, 0) = 0 \quad (A.44)$$

$\Rightarrow$

$$P(m_1+m_0, 0) = F(0, 0) (\frac{\lambda_1}{\mu C})^{m_1+m_0} \quad (A.45)$$

and in the same way

$$P(0, m_2 + m_0) = F(0, 0) \left(\frac{\lambda_2}{\mu C} \right)^{m_2 + m_0} \quad (A.46)$$

Thus

$$F(z_1, 0) = \frac{F(0, 0) \left(1 - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1 + m_0 + 1} \right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1 \right)} \quad (A.47)$$

and

$$F(0, z_2) = \frac{F(0, 0) \left(1 - \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2 + m_0 + 1} \right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2 \right)} \quad (A.48)$$

So, the two equations A.38 and A.39 can be written

$$\begin{aligned} F(z_1, z_2) \left(1 - \frac{\lambda_1}{\mu C} z_1 \right) - \frac{F(0, 0) \left(1 - \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2 + m_0 + 1} \right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2 \right)} + \\ + \frac{\lambda_1}{\mu C} z_1 F_{m_1 + m_0, M} = 0 \end{aligned} \quad (A.49)$$

$$\begin{aligned} F(z_1, z_2) \left(1 - \frac{\lambda_2}{\mu C} z_2 \right) - \frac{F(0, 0) \left(1 - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1 + m_0 + 1} \right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1 \right)} + \\ + \frac{\lambda_2}{\mu C} z_2 F_{m_2 + m_0, M} = 0 \end{aligned} \quad (A.50)$$

To obtain an expression for $F_{m_1 + m_0}$ and $F_{m_2 + m_0}$ we re-write the second equation

$$F(z_1, z_2) \left(1 - \frac{\lambda_2}{\mu C} z_2\right) - F(0, 0) \sum_{k=0}^{m_1+m_3} \left(\frac{\lambda_1}{\mu C} z_1\right)^k +$$

$$+ \frac{\lambda_2}{\mu C} z_2 F_{m_2+m_0, M} = 0 \quad (A.51)$$

and differentiate

$$\left(1 - \frac{\lambda_2}{\mu C} z_2\right) \frac{\partial^{m_1+m_0} F(z_1, z_2)}{\partial z_1^{m_1+m_0}} - F(0, 0) (m_1+m_0)! \left(\frac{\lambda_1}{\mu C}\right)^{m_1+m_0} +$$

$$+ \frac{\lambda_2}{\mu C} z_2 P(m_1+m_0, m_2) z_2^{m_2} (m_1+m_0)! = 0 \quad (A.52)$$

Knowing that

$$z_1^{m_1+m_0} \frac{\partial^{m_1+m_0} F(z_1, z_2)}{\partial z_1^{m_1+m_0}} = (m_1+m_0)! F_{m_1+m_0} \quad (A.53)$$

we find

$$\left(1 - \frac{\lambda_2}{\mu C} z_2\right) (m_1+m_0)! F_{m_1+m_0} \cdot z_1^{-(m_1+m_0)} -$$

$$- F(0, 0) (m_1+m_0)! \left(\frac{\lambda_1}{\mu C}\right)^{m_1+m_0} +$$

$$+ \frac{\lambda_2}{\mu C} z_2 P(m_1+m_0, m_2) z_2^{m_2} (m_1+m_3)! = 0 \quad (A.54)$$

$\Rightarrow$

$$F_{m_1+m_0} = \frac{F(0, 0) \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0}}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} -$$

$$- \frac{\frac{\lambda_2}{\mu C} z_2 P(m_1+m_0, m_2) z_1^{m_1+m_0} z_2^{m_2}}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} \quad (A.55)$$

Again, since the sum of the coefficients before z_2 must be equal to zero, we find

$$F(0,0) \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0} \left(\frac{\lambda_2}{\mu C}\right)^{m_2+1} - P(m_1+m_0, m_2) z_1^{m_1+m_0} \frac{\lambda_2}{\mu C} = 0 \quad (A.56)$$

$\Rightarrow$

$$P(m_1+m_0, m_2) = F(0,0) \left(\frac{\lambda_1}{\mu C}\right)^{m_1+m_0} \left(\frac{\lambda_2}{\mu C}\right)^{m_2} \quad (A.57)$$

and

$$F(m_1, m_2+m_0) = F(0,0) \left(\frac{\lambda_1}{\mu C}\right)^{m_1} \left(\frac{\lambda_2}{\mu C}\right)^{m_2+m_0} \quad (A.58)$$

By multiplying eq A.49 by $\frac{\lambda_2}{\mu C} z_2$ and eq A.50 by $\frac{\lambda_1}{\mu C} z_1$ and subtract the second from the first we can eliminate F_M from the two equations

$$\begin{aligned} & \frac{\lambda_2}{\mu C} z_2 F(z_1, z_2) \left(1 - \frac{\lambda_1}{\mu C} z_1\right) - \frac{\lambda_2}{\mu C} z_2 \frac{F(0,0) \left(1 - \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}\right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} + \\ & + \frac{\lambda_2}{\mu C} z_2 \frac{\lambda_1}{\mu C} z_1 F_{m_1+m_0} + \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 F_M - \\ & - \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 P(m_1+m_0, m_2) z_1^{m_1+m_0} z_2^{m_2} = 0 \quad (A.59) \end{aligned}$$

and

$$\frac{\lambda_1}{\mu C} z_1 F(z_1, z_2) \left(1 - \frac{\lambda_2}{\mu C} z_2\right) - \frac{\lambda_1}{\mu C} z_1 \frac{F(0,0) \left(1 - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1}\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right)} +$$

$$\begin{aligned}
 & + \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 F_{m_2+m_0} + \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 F_M - \\
 & - \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 P(m_1, m_2+m_0) z_1^{m_1} z_2^{m_2+m_0} = 0 \quad (A.60)
 \end{aligned}$$

$\Rightarrow$

$$\begin{aligned}
 & F(z_1, z_2) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1 \right) - \frac{\lambda_2}{\mu C} z_2 \frac{F(0,0) \left(1 - \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+1} \right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2 \right)} + \\
 & + \frac{\lambda_1}{\mu C} z_1 \frac{F(0,0) \left(1 - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+1} \right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1 \right)} + \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 (F_{m_1+m_0} - F_{m_2+m_0}) - \\
 & - \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 (P(m_1+m_0, m_2) z_1^{m_1+m_0} z_2^{m_2} - \\
 & - P(m_1, m_2+m_0) z_1^{m_1} z_2^{m_2+m_0+1}) = 0 \quad (A.61)
 \end{aligned}$$

$\Rightarrow$

$$\begin{aligned}
 & F(z_1, z_2) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1 \right) = \\
 & = F(0,0) \left\{ \frac{\lambda_2}{\mu C} z_2 \frac{\left(1 - \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+1} \right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2 \right)} - \frac{\lambda_1}{\mu C} z_1 \frac{\left(1 - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+1} \right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1 \right)} - \right. \\
 & - \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 F_{m_1+m_0} + \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 P(m_1+m_0, m_2) z_1^{m_1+m_0} z_2^{m_2} + \\
 & + \left. \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 F_{m_2+m_0} - \frac{\lambda_1}{\mu C} z_1 \frac{\lambda_2}{\mu C} z_2 P(m_1, m_2+m_0) z_1^{m_1} z_2^{m_2+m_0} \right\} \\
 & \quad (A.62)
 \end{aligned}$$

As we can express $F_{m_1+m_0}$, $F_{m_2+m_0}$, $P(m_1+m_0, m_2)$ and $P(m_1, m_2+m_0)$ as functions of $F(0,0)$ we have obtained a solution for $F(z_1, z_2)$.

A.2.3 Inversion of the z-transform

In order to retransform $F(z_1, z_2)$ to $P(k_1, k_2)$ we have to rewrite eq A.62 a little.

$$\begin{aligned}
 & F(z_1, z_2) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1 \right) = \\
 & = F(0,0) \left\{ \frac{\frac{\lambda_2}{\mu C} z_2 \left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}\right) - \frac{\lambda_1}{\mu C} z_1 \left(1 - \frac{\lambda_2}{\mu C} z_2\right) \left(1 - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1}\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} \right. \\
 & - \frac{\frac{\lambda_2}{\mu C} z_2 \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1}}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} + \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+2}}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} + \\
 & + \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1}}{\left(\frac{\lambda_1}{\mu C} z_1\right)} + \frac{\frac{\lambda_1}{\mu C} z_1 \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right)} - \\
 & \left. - \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+2} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right)} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1} \right\} \quad (A.63)
 \end{aligned}$$

The first term in the right part of eq A.63 can be written

$$\begin{aligned} & \frac{\left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} - \\ & - \frac{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+2} - \left(1 - \frac{\lambda_2}{\mu C} z_2\right) \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+2}}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} \end{aligned} \quad (A.64)$$

and the two terms

$$\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1} \quad (A.65)$$

as

$$\begin{aligned} & - \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1} \left(\left(\frac{\lambda_2}{\mu C} z_2\right)^{m_0} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}\right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} + \\ & + \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+2} \left(\left(\frac{\lambda_2}{\mu C} z_2\right)^{m_0} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}\right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} \end{aligned} \quad (A.66)$$

Thus eq A.63 can be written

$$\begin{aligned} F(z_1, z_2) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right) &= F(0, 0) \left\{ \frac{\left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} - \right. \\ & \left. - \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1} \left(\left(\frac{\lambda_2}{\mu C} z_2\right)^{m_0} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}\right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} - \right. \end{aligned}$$

$$- \frac{1}{(1 - \frac{\lambda_1}{\mu C} z_1) (1 - \frac{\lambda_2}{\mu C} z_2)} \left(\left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+2} - \right.$$

$$\left. - \left(\frac{\lambda_1}{\mu C} z_1 \right) \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+2} - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+1} + \right.$$

(i)

$$+ \frac{\lambda_2}{\mu C} z_2 \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+2} - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+2} +$$

(v)

$$+ \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+2} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+2} + \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+1} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+2} -$$

(ii) (iii)

$$- \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+2} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+2} + \frac{\lambda_2}{\mu C} z_2 \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+1} -$$

(iv)

$$- \frac{\lambda_2}{\mu C} z_2 \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+2} - \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+1} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+2} +$$

(v) (iii)

$$+ \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+m_0+2} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+2} + \frac{\lambda_1}{\mu C} z_1 \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+2} -$$

(iv) (i)

$$- \frac{\lambda_1}{\mu C} z_1 \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+1} + \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+2} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+1} -$$

$$- \left(\frac{\lambda_1}{\mu C} z_1 \right)^{m_1+2} \left(\frac{\lambda_2}{\mu C} z_2 \right)^{m_2+m_0+2} \} \quad (A.67)$$

$$\Rightarrow F(z_1, z_2) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1 \right) =$$

$$\begin{aligned}
 &= F(0,0) \left\{ \frac{\left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} - \right. \\
 &\quad - \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_0} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right)} - \\
 &\quad - \frac{\left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right) \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1} + \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1}}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} \Big\} \quad (A.68)
 \end{aligned}$$

⇒

$$\begin{aligned}
 &F(z_1, z_2) = \\
 &= F(0,0) \left\{ \frac{\left(1 - \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}\right) - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+m_0+1} + \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} - \right. \\
 &\quad - \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_0} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right)} \Big\} \quad (A.69)
 \end{aligned}$$

⇒

$$\begin{aligned}
 &F(z_1, z_2) = F(0,0) \frac{\left(1 - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1}\right) \left(1 - \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+m_0+1}\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} + \\
 &+ \frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(1 - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}\right)}{\left(1 - \frac{\lambda_1}{\mu C} z_1\right) \left(1 - \frac{\lambda_2}{\mu C} z_2\right)} -
 \end{aligned}$$

$$\frac{\left(\frac{\lambda_1}{\mu C} z_1\right)^{m_1+1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{m_2+1} \left(\left(\frac{\lambda_2}{\mu C} z_2\right)^{m_0} - \left(\frac{\lambda_1}{\mu C} z_1\right)^{m_0}\right)}{\left(1 - \frac{\lambda_2}{\mu C} z_2\right) \left(\frac{\lambda_2}{\mu C} z_2 - \frac{\lambda_1}{\mu C} z_1\right)} \quad (A.70)$$

⇒

$$\begin{aligned} F(z_1, z_2) &= \sum_{k_1=0}^{m_1} \sum_{k_2=0}^{m_2+m_0} \left(\frac{\lambda_1}{\mu C} z_1\right)^{k_1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{k_2} F(0,0) + \\ &+ \sum_{k_1=m_1+1}^{m_1+m_0} \sum_{k_2=0}^{M-k_1} \left(\frac{\lambda_1}{\mu C} z_1\right)^{k_1} \left(\frac{\lambda_2}{\mu C} z_2\right)^{k_2} F(0,0) \end{aligned} \quad (E.71)$$

⇒

$$P(k_1, k_2) = \left(\frac{\lambda_1}{\mu C}\right)^{k_1} \left(\frac{\lambda_2}{\mu C}\right)^{k_2} P(0,0) \quad (A.72)$$

where $P(0,0)$ is given by

$$\sum_{\Omega} P(k_1, k_2) = 1 \quad (A.73)$$

A.3 The steady state distribution corresponding to a node with acknowledgement signals

In our packet switching node model including acknowledgement signals and a potential retransmission we assumed the length of the acknowledgement signal to be exponentially distributed with mean $1/\nu$ bits. Furthermore we assumed the signal to be sent from the adjacent node immediately after the transmission of the appropriate packet being completed.

Furthermore we defined a state $\underline{k}, \underline{j}$ to correspond to the case when we had k_i i-packets waiting for transmission or waiting for transmission to be completed and j_i i-packets being transmitted, but waiting for an acknow-

ledgement signal $\forall i=1,2,\dots,n$. Thus the states $\underline{k}, \underline{j}$ form a Markov chain but not a birth-death process.

A.3.1 The global balance equations

In order to be able to draw the state space and to limit the complexity of our derivations, we start to consider a node with just one outgoing transmission link and we regard all the acknowledgement signals to be positive, i.e. no retransmissions are required.

The technique to solve the steady state probability, i.e. $P(\underline{k}, \underline{j})$, is similar to the one used in A.2. However, this Markov chain, as opposed to that in A.2, is not a birth-death process.

The Markov process contains the following global balance equations.

$$(\lambda + \nu C + \mu C) P(k, j) = \lambda P(k-1, j) + \nu C P(k, j+1) + \mu C P(k+1, j-1) \\ 1 \leq k, j, \quad k+j < M \quad (\text{A.74})$$

$$\lambda P(0, 0) = \nu C P(0, 1) \quad (\text{A.75})$$

$$(\lambda + \mu C) P(k, 0) = \lambda P(k-1, 0) + \nu C P(k, 1) \quad (\text{A.76}) \\ 1 \leq k \leq M-1$$

$$(\lambda + \nu C) P(0, j) = \mu C P(1, j-1) + \nu C P(0, j+1) \quad (\text{A.77}) \\ 1 \leq j \leq M-1$$

$$(\nu C + \mu C) P(k, M-k) = \mu C P(k+1, M-k-1) + \lambda P(k-1, M-k) \\ 1 \leq k \leq M-1 \quad (\text{A.78})$$

$$\nu C P(0, M) = \mu C P(1, M-1) \quad (\text{A.79})$$

$$\mu C P(M, 0) = \lambda P(M-1, 0) \quad (\text{A.80})$$

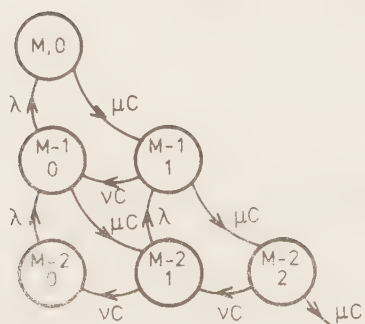


Figure A.5

A.3.2 The z-transform

Define

$$F(z_1, z_2) = \sum_{k=0}^M \sum_{j=0}^{M-k} P(k, j) z_1^k z_2^j ; \quad (A.81)$$

and

$$F(z_1, 0) = \sum_{k=0}^M P(k, 0) z_1^k \quad (A.82)$$

$$F(0, z_2) = \sum_{j=0}^M P(0, j) z_2^j \quad (A.83)$$

$$F_M(z_1, z_2) = \sum_{k=0}^M P(k, M-k) z_1^k z_2^{M-k} \quad (A.84)$$

Multiplying eq A.74 by $z_1^k z_2^j$ and then forming the sum of all such multiplied equations we get

$$\begin{aligned} (\lambda + \mu C + C) \sum_{\Omega_1} \sum P(k, j) z_1^k z_2^j &= \lambda z_1 \sum_{\Omega_2} \sum P(k-1, j) z_1^{k-1} z_2^j + \\ &+ \frac{\nu C}{z_2} \sum_{\Omega_3} \sum P(k, j+1) z_1^k z_2^{j+1} + \frac{\mu C z_2}{z_1} \sum_{\Omega_4} \sum P(k+1, j-1) z_1^{k+1} z_2^{j-1} \end{aligned} \quad (A.85)$$

where the different spaces Ω_i are shown in figure A.6.

The same technique applied to eq A.75 to A.80 gives us

$$\lambda P(0, 0) = \frac{C}{z_2} \sum_{\Omega_5} P(0, 1) z_2 \quad (A.86)$$

$$(\lambda + \mu C) \sum_{\Omega_7} P(k, 0) z_1^k = z_1 \sum_{\Omega_8} P(k, 0) z_1^{k-1} + \frac{\nu C}{z_2} \sum_{\Omega_9} P(k, 1) z_1^k z_2 ; \quad (A.87)$$

$$\begin{aligned}
 & (\lambda + C\nu) \sum_{\Omega_{10}} P(0, j) z_2^j = \\
 & = \frac{\mu C z_2}{z_1} \sum_{\Omega_{11}} P(1, j-1) z_1 z_2^{j-1} + \frac{\nu C}{z_2} \sum_{\Omega_{12}} P(0, j+1) z_2^{j+1} ; \quad (A.88)
 \end{aligned}$$

$$\begin{aligned}
 & (\nu C + \mu C) \sum_{\Omega_{13}} P(k, M-k) z_1^k z_2^{M-k} = \\
 & = \mu C \frac{z_2}{z_1} \sum_{\Omega_{14}} P(k+1, M-k-1) z_1^{k+1} z_2^{M-k-1} + \\
 & + \lambda z_1 \sum_{\Omega_{15}} P(k-1, M-k) z_1^{k-1} z_2^{M-k} ; \quad (A.89)
 \end{aligned}$$

$$\nu C P(0, M) z_2^M = \frac{z_2}{z_1} \sum_{\Omega_{16}} P(1, M-1) z_1 z_2^{M-1} \quad (A.90)$$

$$\mu C P(M, 0) z_1^M = \lambda z_1 \sum_{\Omega_{17}} P(M-1, 0) z_1^{M-1} \quad (A.91)$$

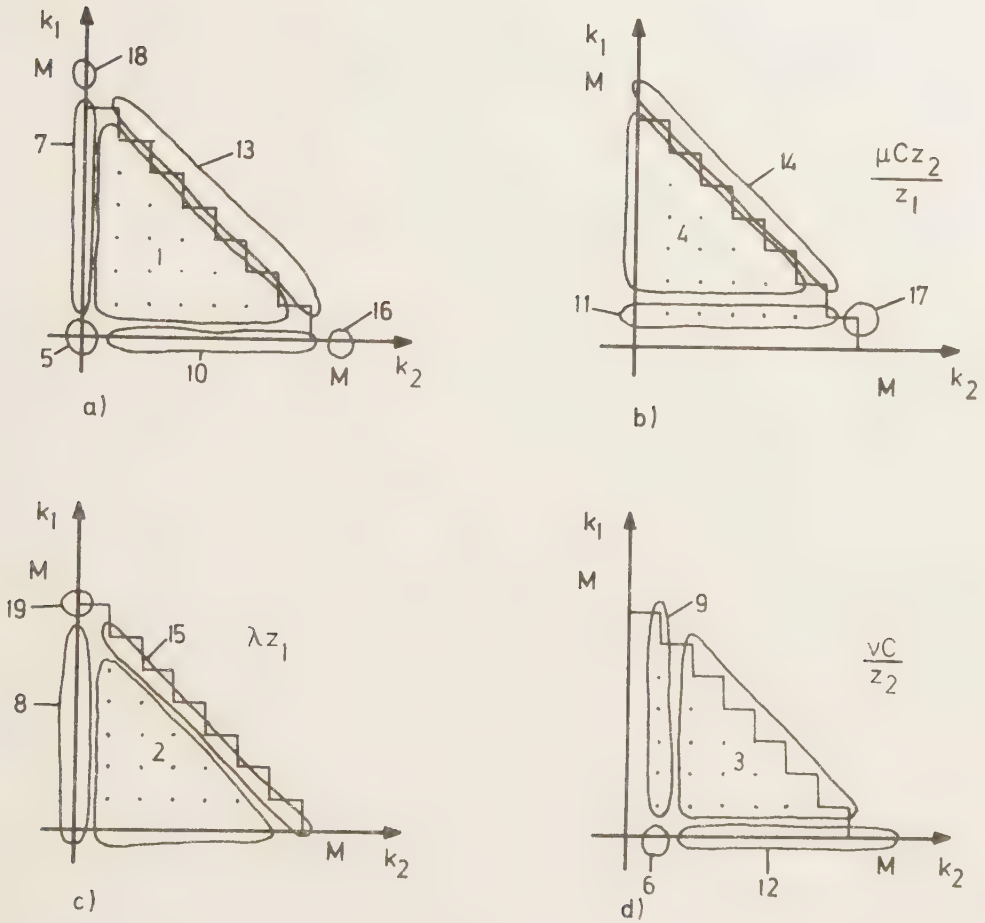


Figure A.6

Addition of the equations A.85 to A.91 gives us

$$\begin{aligned}
 & (\lambda + vC + \mu C) F(z_1, z_2) - (vC + \mu C) P(0, 0) - vC \sum_{k=1}^{M-1} P(k, 0) z_1^k - \\
 & - \mu C \sum_{j=1}^{M-1} P(0, j) z_2^j - \lambda F_M(z_1, z_2) - \mu CP(0, M) z_2^M - vCP(M, 0) z_1^M = \\
 & = \lambda z_1 F(z_1, z_2) - \lambda z_1 F_M(z_1, z_2) + \frac{vC}{z_2} F(z_1, z_2) - \frac{vC}{z_2} F(z_1, 0) +
 \end{aligned}$$

$$+ \frac{\mu C z_2}{z_1} F(z_1, z_2) - \frac{\mu C z_2}{z_1} F(0, z_2) ; \quad (\text{A.92})$$

$\Rightarrow$

$$\begin{aligned} & (\lambda + \nu C + \mu C) F(z_1, z_2) - \lambda F_M(z_1, z_2) - \nu C F(z_1, 0) - \mu C F(0, z_2) = \\ & = \lambda z_1 F(z_1, z_2) - \lambda z_1 F_M(z_1, z_2) + \frac{\nu C}{z_2} F(z_1, z_2) - \frac{\nu C}{z_2} F(z_1, 0) + \\ & + \frac{\mu C z_2}{z_1} F(z_1, z_2) - \frac{\mu C z_2}{z_1} F(0, z_2) \end{aligned} \quad (\text{A.93})$$

$\Rightarrow$

$$\begin{aligned} & F(z_1, z_2) \left(\lambda (1 - z_1) + \nu C \left(1 - \frac{1}{z_2}\right) + \mu C \left(1 - \frac{z_1}{z_2}\right) \right) - \lambda (1 - z_1) F_M(z_1, z_2) - \\ & - \nu C \left(1 - \frac{1}{z_2}\right) F(z_1, 0) - \mu C \left(1 - \frac{z_2}{z_1}\right) F(0, z_2) = 0 \end{aligned} \quad (\text{A.94})$$

We observe that

$$\lambda (1 - z_1) = \lambda (1 - z_2) + \lambda z_2 \left(1 - \frac{z_1}{z_2}\right) \quad (\text{A.95})$$

$$\lambda (1 - z_2) + \nu C \left(1 - \frac{1}{z_2}\right) = \nu C \left(1 - \frac{1}{z_2}\right) \left(1 - \frac{\lambda}{\nu C} z_2\right) \quad (\text{A.96})$$

and

$$\lambda z_2 \left(1 - \frac{z_1}{z_2}\right) + \mu C \left(1 - \frac{z_2}{z_1}\right) = \mu C \left(1 - \frac{z_2}{z_1}\right) \left(1 - \frac{\lambda z_1}{\mu C}\right) \quad (\text{A.97})$$

Thus

$$\begin{aligned} & \nu C \left(1 - \frac{1}{z_2}\right) \{ F(z_1, z_2) \left(1 - \frac{\lambda}{\nu C} z_2\right) - F(z_1, 0) + \frac{\lambda}{\nu C} z_2 F_M(z_1, z_2) \} + \\ & + \mu C \left(1 - \frac{z_2}{z_1}\right) \{ F(z_1, z_2) \left(1 - \frac{\lambda z_1}{\mu C}\right) - F(0, z_2) + \\ & + \frac{\lambda}{\mu C} z_1 F_M(z_1, z_2) \} = 0 \end{aligned} \quad (\text{A.98})$$

As our Markov process is ergodic, there exists a unique steady state distribution [23].

This implies that if

$$F(z_1, z_2) \left(1 - \frac{\lambda}{\nu C} z_2\right) - F(z_1, 0) + \frac{\lambda}{\nu C} z_2 F_M(z_1, z_2) = 0 \quad (\text{A.99})$$

$$F(z_1, z_2) \left(1 - \frac{\lambda}{\mu C} z_1\right) - F(0, z_2) + \frac{\lambda}{\mu C} z_1 F_M(z_1, z_2) = 0 \quad (\text{A.100})$$

give a unique solution $F(z_1, z_2)$, then it is the solution to eq A.98.

Let $z_1 = 0$ in eq A.99

$$F(0, z_2) \left(1 - \frac{\lambda}{\nu C} z_2\right) - F(0, 0) + \frac{\lambda}{\nu C} z_2 P(0, M) z_2^M = 0 \quad (\text{A.101})$$

→

$$F(0, z_2) = \frac{F(0, 0) - \frac{\lambda}{\mu C} z_2^{M+1} P(0, M)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} \quad (\text{A.102})$$

The sum of the coefficients preceeding z_2^{M+1} must be equal to zero

$$\Rightarrow F(0, 0) \left(\frac{\lambda}{\nu C}\right)^{M+1} - \frac{\lambda}{\nu C} \cdot P(0, M) = 0 \quad (\text{A.103})$$

$$\Rightarrow F(0, z_2) = \frac{F(0, 0) \left(1 - \left(\frac{\lambda}{\nu C} z_2\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} \quad (\text{A.104})$$

And in the same way

$$F(z_1, 0) = \frac{F(0, 0) \left(1 - \left(\frac{\lambda}{\mu C} z_1\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\mu C} z_1\right)} \quad (\text{A.105})$$

Thus the two equations A.99 and A.100 can be written

$$F(z_1, z_2) \left(1 - \frac{\lambda}{\nu C} z_2\right) - \frac{F(0,0) \left(1 - \left(\frac{\lambda}{\mu C} z_1\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\mu C} z_1\right)} +$$

$$+ \frac{\lambda}{\nu C} z_2 F_M(z_1, z_2) = 0 \quad (\text{A.106})$$

$$F(z_1, z_2) \left(1 - \frac{\lambda}{\mu C} z_1\right) - \frac{F(0,0) \left(1 - \left(\frac{\lambda}{\nu C} z_2\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} +$$

$$+ \frac{\lambda}{\mu C} z_1 F_M(z_1, z_2) = 0 \quad (\text{A.107})$$

If we multiply eq A.106 by $\frac{\lambda}{\mu C} z_1$ and eq A.107 by $\frac{\lambda}{\nu C} z_2$ and subtract eq A.106 from A.107 we obtain

$$F(z_1, z_2) \left(\frac{\lambda}{\nu C} z_2 - \frac{\lambda}{\mu C} z_1\right) =$$

$$= \frac{\frac{\lambda}{\nu C} z_2 F(0,0) \left(1 - \left(\frac{\lambda}{\nu C} z_2\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} - \frac{\frac{\lambda}{\mu C} z_1 F(0,0) \left(1 - \left(\frac{\lambda}{\mu C} z_1\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\mu C} z_1\right)};$$

$$(\text{A.108})$$

⇒

$$F(z_1, z_2) \left(\frac{\lambda}{\nu C} z_2 - \frac{\lambda}{\mu C} z_1\right) =$$

$$= \frac{\left(\frac{\lambda}{\nu C} z_2 - \frac{\lambda}{\mu C} z_1\right) \left(1 - \left(\frac{\lambda}{\mu C} z_1\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\mu C} z_1\right) \left(1 - \frac{\lambda}{\nu C} z_2\right)} - \frac{\frac{\lambda}{\nu C} z_2 \left(\left(\frac{\lambda}{\nu C} z_2\right)^{M+1} - \left(\frac{\lambda}{\mu C} z_1\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)};$$

$$(\text{A.109})$$

⇒

$$F(z_1, z_2) = \frac{F(0,0) \left(1 - \frac{\lambda}{\mu C} z_1\right)^{M+1}}{\left(1 - \frac{\lambda}{\mu C} z_1\right) \left(1 - \frac{\lambda}{\nu C} z_2\right)} -$$

$$- \frac{\left(\frac{\lambda}{\mu C} z_2\right)^{M+1} \left(1 - \left(\frac{\nu C z_1}{\mu C z_2}\right)^{M+1}\right)}{\left(1 - \frac{\nu C z_1}{\mu C z_2}\right) \left(1 - \frac{\lambda}{\nu C} z_2\right)} \cdot F(0,0) \quad (A.110)$$

⇒

$$\begin{aligned} F(z_1, z_2) &= \frac{F(0,0)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} \left\{ \sum_{k=0}^M \left(\frac{\lambda}{\mu C} z_1\right)^k - \left(\frac{\lambda}{\nu C} z_2\right)^{M+1} \cdot \sum_{k=0}^M \left(\frac{\nu C z_1}{\mu C z_2}\right)^k \right\} = \\ &= F(0,0) \sum_{k=0}^M \left\{ \frac{\left(\frac{\lambda}{\mu C} z_1\right)^k}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} - \frac{\left(\frac{\lambda}{\mu C} z_1\right)^k \left(\frac{\lambda}{\nu C} z_2\right)^{M-k+1}}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} \right\} = \\ &= F(0,0) \sum_{k=0}^M \left(\frac{\lambda}{\mu C} z_1\right)^k \frac{\left(1 - \left(\frac{\lambda}{\nu C} z_2\right)^{M-k+1}\right)}{\left(1 - \frac{\lambda}{\nu C} z_2\right)} = \\ &= F(0,0) \sum_{k=0}^M \left(\frac{\lambda}{\mu C} z_1\right)^k \sum_{j=0}^{M-k} \left(\frac{\lambda}{\nu C} z_2\right)^j ; \quad (A.111) \end{aligned}$$

⇒

$$P(k, j) = \left(\frac{\lambda}{\mu C}\right)^k \left(\frac{\lambda}{\nu C}\right)^j P(0,0) \quad (A.112)$$

where $P(0,0)$ is obtained from

$$\sum_{k=0}^M \sum_{j=0}^{M-k} P(k, j) = 1 \quad (A.113)$$

⇒

$$P(0,0) = \frac{\left(1 - \left(\frac{\lambda}{\mu C}\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\nu C}\right) \left(1 - \frac{\lambda}{\mu C}\right)} - \frac{\left(\frac{\lambda}{\nu C}\right)^{M+1} \left(1 - \left(\frac{\nu}{\mu C}\right)^{M+1}\right)}{\left(1 - \frac{\lambda}{\nu C}\right) \left(1 - \frac{\nu}{\mu C}\right)} \quad (A.114)$$

These derivations could be extended to a node with n outgoing transmission links with the steady state probability

$$P(\underline{k}, \underline{j}) = \prod_{i=1}^n \left(\frac{\lambda_i}{\mu C} \right)^{k_i} \left(\frac{\lambda_i}{\nu C} \right)^{j_i} P(\underline{0}) \quad (\text{A.115})$$

A.4 Acknowledgement signals and retransmission

In this section we extend our model in A.3.1 and assume with probability α an acknowledgement signal to be negative and thus with probability α requiring a retransmission of the appropriate packet.

Again we restrict our derivations to a node with just one outgoing transmission link and now also with an unlimited number of buffers.

A.4.1 The global balance equations

Figure A.7 shows the corresponding state space. From the state space we list the global balance equations

$$\begin{aligned} (\lambda + \nu C + \mu C) P(k, l) &= \lambda P(k-1, l) + \mu C P(k+1, l-1) + \\ &+ (1-\alpha) \nu C P(k, l+1) + \alpha \nu C P(k-1, l+1) \quad k > 0, \quad l > 0 \end{aligned} \quad (\text{A.116})$$

$$\begin{aligned} (\lambda + C\mu) P(k, 0) &= \lambda P(k-1, 0) + (1-\alpha) \nu C P(k, 1) + \alpha \nu C P(k-1, 1) \\ l = 0 \quad k > 0 \end{aligned} \quad (\text{A.117})$$

$$\begin{aligned} (\lambda + \nu C) P(0, l) &= \mu C P(1, l-1) + (1-\alpha) \nu C P(0, l+1) \\ k = 0, \quad l > 0 \end{aligned} \quad (\text{A.118})$$

$$\lambda P(0, 0) = (1-\alpha) \nu C P(0, 1) \quad (\text{A.119})$$

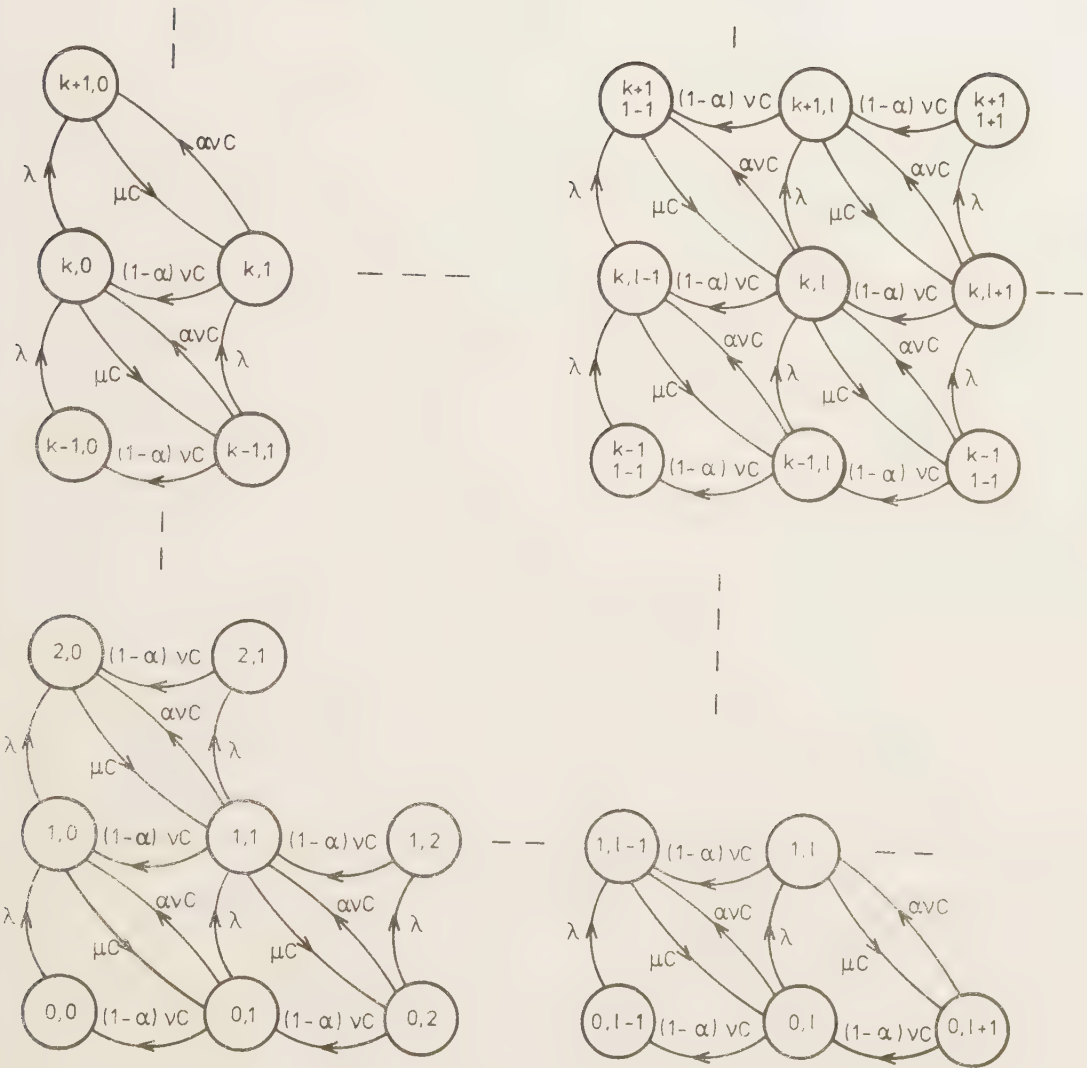


Figure A.7

A.4.2 The z-transform

We define a z-transform

$$F(z_1, z_2) = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} P(k, l) z_1^k z_2^l \quad (\text{A.120})$$

Equations A.116 to A.119 lead to

$$\begin{aligned} (\lambda + \nu C + \mu C) \sum_{k=1}^{\infty} \sum_{l=1}^{\infty} P(k, l) z_1^k z_2^l &= \lambda z_1 \sum_{k=0}^{\infty} \sum_{l=1}^{\infty} P(k, l) z_1^k z_2^l + \\ &+ \frac{\mu C z_2}{z_1} \sum_{k=2}^{\infty} \sum_{l=0}^{\infty} P(k, l) z_1^k z_2^l + \frac{(1-\alpha)\nu C}{z_2} \sum_{k=1}^{\infty} \sum_{l=2}^{\infty} P(k, l) z_1^k z_2^l + \\ &+ \frac{\alpha \nu C z_1}{z_2} \sum_{k=0}^{\infty} \sum_{l=2}^{\infty} P(k, l) z_1^k z_2^l ; \end{aligned} \quad (\text{A.121})$$

$$\begin{aligned} (\lambda + \mu C) \sum_{k=1}^{\infty} P(k, 0) z_1^k &= \lambda z_1 \sum_{k=0}^{\infty} P(k, 0) z_1^k + \\ &+ \frac{(1-\alpha)\nu C}{z_2} \sum_{k=1}^{\infty} P(k, 1) z_1^k z_2 + \frac{\alpha \nu C z_1}{z_2} \sum_{k=0}^{\infty} P(k, 1) z_1^k z_2 ; \end{aligned} \quad (\text{A.122})$$

$$\begin{aligned} (\lambda + \nu C) \sum_{l=1}^{\infty} P(0, l) z_2^l &= \\ &= \frac{\mu C z_2}{z_1} \sum_{l=0}^{\infty} P(1, l) z_1 z_2^l + \frac{(1-\alpha)\nu C}{z_2} \sum_{l=2}^{\infty} P(0, l) z_2^l ; \end{aligned} \quad (\text{A.123})$$

$$\lambda P(0, 0) = \frac{(1-\alpha)\nu C}{z_2} P(0, 1) z_2 ; \quad (\text{A.124})$$

If we sum up these four equations we obtain an equation with the following left part

$$\begin{aligned}
 & (\lambda + \nu C + \mu C) F(z_1, z_2) - \nu C \sum_{k=1}^{\infty} P(k, 0) z_1^k - \mu C \sum_{l=1}^{\infty} P(0, l) z_2^l - \\
 & - (\nu C + \mu C) P(0, 0) ; \quad (A.125)
 \end{aligned}$$

⇒

$$(\lambda + \nu C + \mu C) F(z_1, z_2) - \nu C F(z_1, 0) - \mu C F(0, z_2) ; \quad (A.126)$$

The right part of the equation could be divided into four parts

$$\begin{aligned}
 & \lambda z_1 F(z_1, z_2) \\
 & \frac{\mu C z_2}{z_1} (F(z_1, z_2) - F(0, z_2)) \\
 & \frac{(1-\alpha) \nu C}{z_2} (F(z_1, z_2) - F(z_1, 0)) \\
 & \frac{\alpha \nu C z_1}{z_2} (F(z_1, z_2) - F(z_1, 0))
 \end{aligned}$$

Thus we write the equation as follows

$$\begin{aligned}
 & (\lambda + \nu C + \mu C) F(z_1, z_2) - \nu C F(z_1, 0) - \mu C F(0, z_2) = \\
 & = \lambda z_1 F(z_1, z_2) + \frac{\mu C z_2}{z_1} (F(z_1, z_2) - F(0, z_2)) + \\
 & + \frac{(1-\alpha) \nu C}{z_2} (F(z_1, z_2) - F(z_1, 0)) + \frac{\alpha \nu C z_1}{z_2} (F(z_1, z_2) - F(z_1, 0)) \quad (A.127)
 \end{aligned}$$

⇒

$$\begin{aligned}
 & F(z_1, z_2) \left(\lambda (1 - z_1) - \lambda \left(1 - \frac{z_2}{z_1} \right) + \nu C \left(1 - \frac{1-\alpha}{z_2} - \frac{\alpha z_1}{z_2} \right) \right) - \\
 & - \nu C \left(1 - \frac{1-\alpha}{z_2} - \frac{\alpha z_1}{z_2} \right) F(z_1, 0) - \mu C \left(1 - \frac{z_2}{z_1} \right) F(0, z_2) = 0 \quad (A.128)
 \end{aligned}$$

Writing $(1-z_1)$ as

$$\frac{\lambda}{1-\alpha} (z_2 - z_1) + \lambda \left(1 - \frac{z_2}{1-\alpha}\right) - \lambda z_1 \left(1 - \frac{1}{1-\alpha}\right)$$

and observing that

$$\frac{\lambda}{1-\alpha} (z_2 - z_1) + \mu C \left(1 - \frac{z_2}{z_1}\right) = \mu C \left(1 - \frac{z_2}{z_1}\right) \left(1 - \frac{\lambda}{\mu C (1-\alpha)} z_1\right) \quad (\text{A.129})$$

and

$$\begin{aligned} & \lambda \left(\left(1 - \frac{z_2}{1-\alpha}\right) - z_1 \left(1 - \frac{1}{1-\alpha}\right) \right) + \nu C \left(1 - \frac{1-\alpha}{z_2} - \frac{\alpha z_1}{z_2}\right) = \\ & = \nu C \left(1 - \frac{1-\alpha}{z_2} - \frac{\alpha z_1}{z_2}\right) \left(1 - \frac{\lambda}{\nu C (1-\alpha)} z_2\right) \end{aligned} \quad (\text{A.130})$$

implies that eq A.118 could be written as

$$\begin{aligned} & \mu C \left(1 - \frac{z_2}{z_1}\right) (F(z_1, z_2) \left(1 - \frac{\lambda}{\mu C (1-\alpha)} z_1\right) - F(0, z_2)) - \\ & - \nu C \left(1 - \frac{1-\alpha}{z_2} - \frac{\alpha z_1}{z_2}\right) (F(z_1, z_2) \left(1 - \frac{\lambda}{\nu C (1-\alpha)} z_2\right) - F(z_1, 0)) = 0 \end{aligned} \quad (\text{A.131})$$

As earlier we claim

$$F(z_1, z_2) \left(1 - \frac{\lambda}{\mu C (1-\alpha)} z_1\right) - F(0, z_2) = 0 \quad (\text{A.132})$$

$$F(z_1, z_2) \left(1 - \frac{\lambda}{\nu C (1-\alpha)} z_2\right) - F(z_1, 0) = 0 \quad (\text{A.133})$$

Let $z_2 = 0$ in eq A.132 $\Rightarrow$

$$F(z_1, 0) = \frac{F(0, 0)}{\left(1 - \frac{\lambda}{\mu C (1-\alpha)} z_1\right)} ; \quad (\text{A.134})$$

and $z_1 = 0$ in eq A.133 $\Rightarrow$

$$F(0, z_2) = \frac{F(0, 0)}{(1 - \frac{\lambda}{\nu C(1-\alpha)} z_2)} ; \quad (\text{A.135})$$

Thus we obtain

$$F(z_1, z_2) = \frac{F(0, 0)}{(1 - \frac{\lambda}{\mu C(1-\alpha)} z_1)(1 - \frac{\lambda}{\nu C(1-\alpha)} z_2)} ; \quad (\text{A.136})$$

$\Rightarrow$

$$P(k, 1) = (\frac{\lambda}{\mu C(1-\alpha)})^k (\frac{\lambda}{\nu C(1-\alpha)})^1 P(0, 0) ; \quad (\text{A.137})$$

where

$$P(0, 0) = (1 - \frac{\lambda}{\mu C(1-\alpha)}) (1 - \frac{\lambda}{\nu C(1-\alpha)}) ; \quad (\text{A.138})$$

We have derived an expression for the probability of finding k packets waiting for transmission or for transmission to be completed and j packets waiting for an acknowledgement signal. The probability of finding k packets in the buffers

$$P(k) = \sum_{l=0}^k P(k-l, 1) ; \quad (\text{A.139})$$

$$P(k) = \sum_{l=0}^k (\frac{\lambda}{\mu C(1-\alpha)})^{k-l} (\frac{\lambda}{\nu C(1-\alpha)})^l P(0, 0)$$

$$= \begin{cases} \frac{(\frac{\lambda}{\mu C(1-\alpha)})^{k+1} - (\frac{\lambda}{\nu C(1-\alpha)})^{k+1}}{\frac{\lambda}{\mu C(1-\alpha)} - \frac{\lambda}{\nu C(1-\alpha)}} (1 - \frac{\lambda}{\mu C(1-\alpha)}) (1 - \frac{\lambda}{\nu C(1-\alpha)}) & \mu C \neq \nu C \\ (k+1) (\frac{\lambda}{\mu C(1-\alpha)})^k (1 - \frac{\lambda}{\mu C(1-\alpha)})^2 & \mu C = \nu C \end{cases} \quad (\text{A.140})$$

To derive the steady state distribution, when we have a limited number of buffers, we use the same technique as in the unlimited case.

This method leads to

$$P(k, l) = \left(\frac{\lambda}{\mu C (1-\alpha)} \right)^k \left(\frac{\lambda}{\nu C (1-\alpha)} \right)^l P(0, 0) \quad (\text{A.141})$$

where

$$P(0, 0) = \frac{\left(1 - \frac{\lambda}{\mu C (1-\alpha)}\right)^{M+1} \left(1 - \frac{\lambda}{\nu C (1-\alpha)}\right)^{M+1}}{\left(1 - \left(\frac{\lambda}{\mu C (1-\alpha)}\right)^{M+1}\right) \left(1 - \left(\frac{\lambda}{\nu C (1-\alpha)}\right)^{M+1}\right)}; \quad (\text{A.142})$$

We assume there are M buffers in the node.

The solution above could be checked in the global balance equations for our two-dimensional Markov process gained from the state space (see figure A.8).

$$\begin{aligned} (\lambda + \nu C + \mu C) P(k, l) &= \lambda P(k-1, l) + \mu C P(k+1, l-1) + \\ &+ (1-\alpha) \nu C P(k, l+1) + \alpha \nu C P(k-1, l+1) \end{aligned} \quad (\text{A.143})$$

$k > 0, \quad l > 0, \quad k+1 < M$

$$\begin{aligned} (\lambda + \mu C) P(k, 0) &= \lambda P(k-1, 0) + (1-\alpha) \nu C P(k, 1) + \alpha \nu C P(k-1, 1) \\ l &= 0 \quad 0 < k < M \end{aligned} \quad (\text{A.144})$$

$$\begin{aligned} (\lambda + \nu C) P(0, l) &= \mu C P(1, l-1) + (1-\alpha) \nu C P(0, l+1) \\ k &= 0 \quad 0 < l < M \end{aligned} \quad (\text{A.145})$$

$$\begin{aligned} (\nu C + \mu C) P(k, M-k) &= \lambda P(k-1, M-k) + \mu C P(k+1, M-k-1) + \\ &+ \alpha \nu C P(k-1, M-k+1); \quad 1 < l < M \end{aligned} \quad (\text{A.146})$$

$$\mu C P(M, 0) = \lambda P(M-1, 0) + \alpha \nu C P(M-1, 1); \quad (\text{A.147})$$

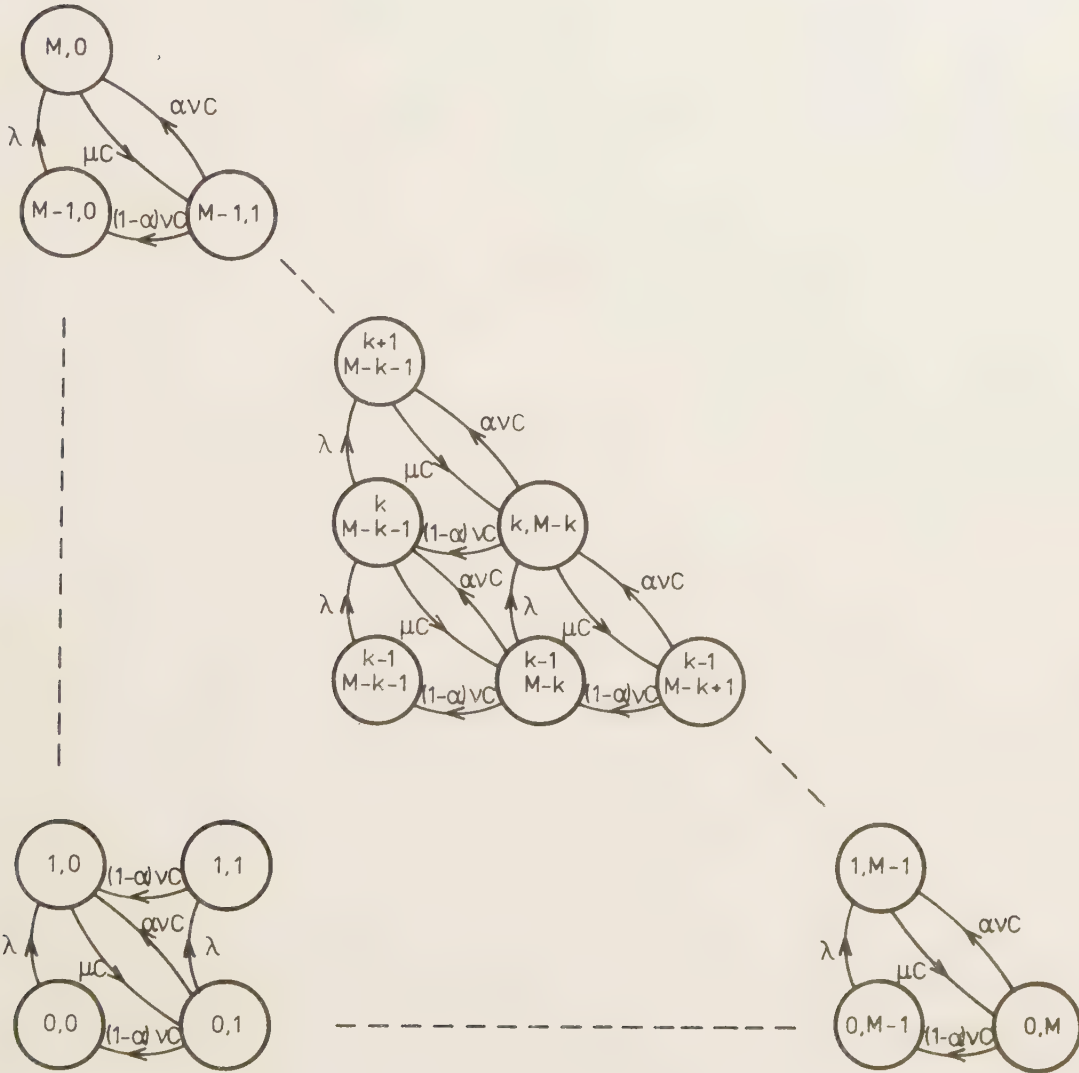


Figure A.8

$$\nu CP(0, M) = \mu CP(1, M-1); \quad (A.148)$$

$$\lambda P(0, 0) = (1-\alpha) \nu CP(0, 1); \quad (A.149)$$

These global balance equations are fulfilled by eq A.141, so eq A.141 is our steady state distribution.

We have found the probability of finding k packets in the buffers

$$P(k) = \frac{\left(\frac{\lambda}{\mu C(1-\alpha)}\right)^{k+1} - \left(\frac{\lambda}{\nu C(1-\alpha)}\right)^{k+1}}{\frac{\lambda}{\mu C(1-\alpha)} - \frac{\lambda}{\nu C(1-\alpha)}} \cdot P(0, 0) \quad \mu C \neq \nu C \quad (A.150)$$

When we have n outgoing transmission links we assume packets bound for transmission link i , i - packets, to arrive according to a Poisson process with rate λ_i , $i = 1, 2, \dots, n$. Thus the steady state probability $P(\underline{k}, \underline{l})$, where $\underline{k} = (k_1, k_2, \dots, k_n)$ and $\underline{l} = (l_1, l_2, \dots, l_n)$, could be written

$P(\underline{k}, \underline{l}) = P(k_i \text{ packets in the buffers waiting for transmission over transmission link } i \text{ or for transmission on link } i \text{ to be completed and } j_i \text{ packets in the buffers waiting for an acknowledgement signal corresponding to link } i, i = 1, 2, \dots, n)$

$$P(\underline{k}, \underline{l}) = \prod_{i=1}^n \left(\frac{\lambda_i}{\mu C(1-\alpha)}\right)^{k_i} \prod_{i=1}^n \left(\frac{\lambda_i}{\nu C(1-\alpha)}\right)^{l_i} P(\underline{0}) \quad (A.151)$$

where $\underline{0} = (0, 0, \dots, 0)$

$P(\underline{0})$ is derived from the normalization

$$\sum_{\Omega} P(\underline{k}, \underline{l}) = 1 \quad (A.152)$$

Thus the principle structure of $P(\underline{k}, \underline{l})$ does not depend on different restricted buffer sharing policies.

Let $P(\underline{k}) = P(k_i \text{ packets in the buffers corresponding to transmission link } i, i = 1, 2, \dots, n)$

If $n = 2$ then

$$P(k_1, k_2) = \frac{(\rho_{11}^{k_1+1} - \rho_{12}^{k_1+1})(\rho_{21}^{k_2+1} - \rho_{22}^{k_2+1})}{(\rho_{11} - \rho_{12})(\rho_{21} - \rho_{22})} \cdot P(0, 0) \quad (\text{A.153})$$

APPENDIX B

B.1 Introduction

In chapter 6, where we applied restricted buffer sharing policies to our packet switching node model, we found that total utilization (or throughput) as a function of offered load decreased under the CS policy. Under the SMA policy total utilization formed on the contrary a monotone non-decreasing function of offered load.

In section B.2 and B.3 we will derive the expressions for the asymptotic behaviour of utilization when offered load increases dramatically. The expressions are derived for a model with two outgoing transmission links and under the unbalanced load pattern.

B.2 Utilization under the CS policy

Under the CS policy we found, when $\rho_1 \neq \rho_2 \neq 1$, the normalization constant $P(0,0)$ to be given by the expression

$$\frac{1}{P(0,0)} = \frac{1}{(1-\rho_2)} \left(\frac{1-\rho_1^{M+1}}{1-\rho_1} - \frac{\rho_2 (\rho_2^{M+1} - \rho_1^{M+1})}{(\rho_2 - \rho_1)} \right); \quad (B.1)$$

$\Rightarrow$

$$P(0,0) = \frac{(1-\rho_2)(1-\rho_1)(\rho_2 - \rho_1)}{(\rho_2 - \rho_1)(1-\rho_1^{M+1}) - \rho_2(1-\rho_1)(\rho_2^{M+1} - \rho_1^{M+1})} \quad (B.2)$$

Furthermore we found the two blocking probabilities B_1 and B_2 to be equal and we used the notation B for them. We found

$$B = \frac{\rho_2^{M+1} - \rho_1^{M+1}}{\rho_2 - \rho_1} P(0,0); \quad (B.3)$$

Thus

$$B = \frac{(\rho_2^{M+1} - \rho_1^{M+1})(1-\rho_1)(1-\rho_2)}{(\rho_2 - \rho_1)(1-\rho_1^{M+1}) - \rho_2(1-\rho_1)(\rho_2^{M+1} - \rho_1^{M+1})}; \quad (B.4)$$

which leads to

$$1 - B = \frac{(\rho_2 - \rho_1)(1-\rho_1^{M+1}) - (1-\rho_1)(\rho_2^{M+1} - \rho_1^{M+1})}{(\rho_2 - \rho_1)(1-\rho_1^{M+1}) - \rho_2(1-\rho_1)(\rho_2^{M+1} - \rho_1^{M+1})}; \quad (B.5)$$

Utilization of transmission link i ($i = 1, 2$)

$$= \rho_i(1-B) \quad (B.6)$$

If we now, under our unbalanced load situation, hold ρ_1 constant and let ρ_2 increase, the utilization of transmission link i is equal to

$$\lim_{\rho_2 \rightarrow \infty} \rho_i(1-B) \quad (B.7)$$

In the expression for $1-B$ we find that the highest power of ρ_2 is $M+1$ in the denominator and is preceded by the coefficient $-(1-\rho_1)$. The highest power of ρ_2 in the numerator is $M+2$ and the corresponding coefficient is $-(1-\rho_1)$.

Thus

$$\lim_{\rho_2 \rightarrow \infty} \rho_1(1-B) = 0 \quad (B.8)$$

and

$$\lim_{\rho_2 \rightarrow \infty} \rho_2(1-B) = 1 \quad (B.9)$$

B.3 Utilization under the SMA policy

Under the SMA policy and when ρ_1 , ρ_2 and 1 are all distinct, we can write the expression for the normalization constant as

$$\begin{aligned} \frac{1}{P(0,0)} &= \frac{1}{(1-\rho_1)(1-\rho_2)(1-\rho_2)} [(\rho_1-\rho_2)(1-\rho_1)^{m_1+m_0+1}(1-\rho_2)^{m_2+1} + \\ &+ (\rho_1-\rho_2)(1-\rho_2)^{m_0}\rho_2^{m_2+1} - (1-\rho_2)\rho_1^{m_1+1}\rho_2^{m_2+1}(\rho_1-\rho_2)^{m_0}] \end{aligned} \quad (B.10)$$

The blocking probability for 1-packets is

$$\begin{aligned} B_1 &= \frac{P(0,0)}{(1-\rho_2)(\rho_1-\rho_2)} [(\rho_1-\rho_2)\rho_1^{m_1+m_0}(1-\rho_2)^{m_2+1} + \\ &+ (1-\rho_2)\rho_1^{m_1}\rho_2^{m_2+1}(\rho_1-\rho_2)^{m_0}] \end{aligned} \quad (B.11)$$

If we write

$$B_1 = \frac{T_1}{N_1} \quad (B.12)$$

then

$$\begin{aligned} T_1 &= (1-\rho_1)(\rho_1-\rho_2)\rho_1^{m_1+m_0}(1-\rho_2)^{m_2+1} + \\ &+ (1-\rho_1)(1-\rho_2)\rho_1^{m_1}\rho_2^{m_2+1}(\rho_1-\rho_2)^{m_0} \end{aligned} \quad (B.13)$$

and

$$N_1 = (\rho_1-\rho_2)(1-\rho_1)^{m_1+m_0+1}(1-\rho_2)^{m_2+1} +$$

$$+ (\rho_1 - \rho_2) (1 - \rho_2)^{m_3} \rho_2^{m_2+1} - (1 - \rho_2) \rho_1^{m_1+1} \rho_2^{m_2+1} (\rho_1^{m_0} - \rho_2^{m_0}) \quad (B.14)$$

Thus utilization of transmission link 1 becomes

$$\rho_1 (1 - B_1) = \rho_1 \left(\frac{N_1 - T_1}{N_1} \right) \quad (B.15)$$

Here we find

$$\begin{aligned} \rho_1 (N_1 - T_1) &= \rho_1 (\rho_1 - \rho_2) (1 - \rho_1)^{m_1+m_0+1} (1 - \rho_2)^{m_2+1} + \\ &+ \rho_1 (\rho_1 - \rho_2) (1 - \rho_2)^{m_0} \rho_2^{m_2+1} - \\ &- \rho_1 (1 - \rho_2) \rho_1^{m_1+1} \rho_2^{m_2+1} (\rho_1^{m_0} - \rho_2^{m_0}) - \\ &- \rho_1 (1 - \rho_1) (\rho_1 - \rho_2) \rho_1^{m_1+m_0} (1 - \rho_1)^{m_2+1} - \\ &- \rho_1 (1 - \rho_1) (1 - \rho_2) \rho_1^{m_1} \rho_2^{m_2+1} (\rho_1^{m_0} - \rho_2^{m_0}) \end{aligned} \quad (B.16)$$

$\Rightarrow$

$$\begin{aligned} \rho_1 (N_1 - T_1) &= \rho_1 (1 - \rho_1)^{m_1+m_0+1} (\rho_1 - \rho_2 - \rho_1 \rho_2^{m_2+1} + \rho_2^{m_2+2}) + \\ &+ \rho_1^2 \rho_2^{m_2+1} - \rho_1^2 \rho_2^{m_2+m_0+1} - \rho_1 \rho_2^{m_2+2} + \rho_1 \rho_2^{m_2+m_0+2} - \\ &- \rho_1^{m_1+m_0+2} \rho_2^{m_2+1} + \rho_1^{m_1+2} \rho_2^{m_2+m_0+1} + \rho_1^{m_1+m_0+2} \rho_2^{m_2+2} - \\ &- \rho_1^{m_1+2} \rho_2^{m_2+m_0+2} - \rho_1 (1 - \rho_1) \rho_1^{m_1+m_0+1} + \\ &+ \rho_1 (1 - \rho_1) \rho_1^{m_1+m_0+1} \rho_2^{m_2+1} + \end{aligned}$$

$$\begin{aligned}
 & + \rho_1 (1-\rho_1) \rho_1^{m_1+m_0} \rho_2 - \rho_1 (1-\rho_1) \rho_1^{m_1+m_0} \rho_2^{m_1+2} - \\
 & - \rho_1 (1-\rho_1) \rho_1^{m_1+m_0} \rho_2^{m_2+1} + \rho_1 (1-\rho_1) \rho_1^{m_1} - \\
 & - \rho_1 (1-\rho_1) \rho_1^{m_1} \rho_2^{m_2+m_0+2} + \rho_1 (1-\rho_1) \rho_1^{m_1+m_0} \rho_2^{m_2+2}
 \end{aligned} \tag{B.17}$$

Moreover

$$\begin{aligned}
 N_1 = & (1-\rho_1^{m_1+m_0+1}) (\rho_1 - \rho_2 - \rho_1 \rho_2^{m_2+1} + \rho_2^{m_2+2}) + \\
 & + \rho_1 \rho_2^{m_2+1} - \rho_1 \rho_2^{m_2+m_0+1} - \rho_2^{m_2+2} + \rho_2^{m_2+m_0+2} - \\
 & - \rho_1^{m_1+m_0+1} \rho_2^{m_2+1} + \rho_1^{m_1+1} \rho_2^{m_2+m_0+1} + \rho_1^{m_1+m_0+1} \rho_2^{m_2+2} - \\
 & - \rho_1^{m_1+1} \rho_2^{m_2+m_0+2}
 \end{aligned} \tag{B.18}$$

Utilization of transmission line 1 when $\rho_2 \rightarrow \infty$

$$= \lim_{\rho_2 \rightarrow \infty} \rho_1 \left(\frac{N_1 - T_1}{N_1} \right) \tag{B.19}$$

Let us first regard the case when $m_0 > 0$. In both the numerator and the denominator the highest power of ρ_2 is m_2+m_0+2 .

The coefficient for $\rho_2^{m_2+m_0+2}$ in the denominator is

$$\rho_1 (1-\rho_1^{m_1}) \tag{B.20}$$

The coefficient for $\rho_2^{m_2+m_0+2}$ in the numerator is

$$1 - \rho_1^{m_1+1} \tag{B.21}$$

Thus

$$\lim_{\rho_2 \rightarrow \infty} \rho_1 (1-B_1) = \frac{\rho_1 (1-\rho_1)^{m_1}}{1-\rho_1^{m_1+1}} \quad (B.22)$$

Now let $m_0 = 0$

In both the numerator and the denominator the highest power of ρ_2 is m_2+2 .

The coefficients in front of $\rho_2^{m_2+2}$ in the denominator are

$$\begin{aligned} & \rho_1 (1-\rho_1)^{m_1+1} - \rho_1 + \rho_1 + \rho_1^{m_1+2} - \rho_1 (1-\rho_1) \rho_1^{m_1} - \\ & - \rho_1 (1-\rho_1) \rho_1^{m_1} + \rho_1 (1-\rho_1) \rho_1^{m_1} \end{aligned} \quad (B.23)$$

which is equal to

$$\rho_1 (1-\rho_1)^{m_1} \quad (B.24)$$

The coefficient in front of $\rho_2^{m_2+2}$ in the numerator is

$$(1-\rho_1)^{m_1+1} \quad (B.25)$$

which gives the same limit for utilization of transmission link 1 as for $m_3 > 0$.

Thus independent of whether $m_0 = 0$ or not, 1-packets can only use their own m_1 dedicated buffers and will find their part of the system behaving like an "M/M/1· m_1 buffers" system.

Utilization of transmission link 2 =

$$= \rho_2 (1 - B_2) \quad (B.26)$$

where

$$B_2 = \frac{P(0,0)}{(1-\rho_1)(\rho_1-\rho_2)} (\rho_1-\rho_2) \rho_2^{m_2+m_0} (1-\rho_1)^{m_1+1} + \\ + (1-\rho_1) \rho_2^{m_2} \rho_1^{m_1+1} (\rho_1-\rho_2)^{m_0} \quad (B.27)$$

$$\Rightarrow B_2 = \frac{T_2}{N_2} \quad (B.28)$$

$$T_2 = (1-\rho_2) (\rho_1-\rho_2) \rho_2^{m_2+m_0} (1-\rho_1)^{m_1+1} + \\ + (1-\rho_2) (1-\rho_1) \rho_2^{m_2} \rho_1^{m_1+1} (\rho_1-\rho_2)^{m_0} \quad (B.29)$$

$$N_2 = (\rho_1-\rho_2) (1-\rho_1)^{m_1+m_0+1} (1-\rho_2)^{m_2+1} + \\ + (\rho_1-\rho_2) (1-\rho_2)^{m_0} \rho_2^{m_2+1} - (1-\rho_2) \rho_1^{m_1+1} \rho_2^{m_2+1} (\rho_1-\rho_2)^{m_0} \quad (B.30)$$

Thus the utilization of transmission link 2 equals

$$\rho_2 \left(\frac{N_2 - T_2}{N_2} \right) \quad (B.31)$$

$$\rho_2 (N_2 - T_2) = \rho_2 (\rho_1-\rho_2) (1-\rho_1)^{m_1+m_0+1} (1-\rho_2)^{m_2+1} + \\ + \rho_2 (\rho_1-\rho_2) (1-\rho_2)^{m_0} \rho_2^{m_2+1} - \rho_2 (1-\rho_2) \rho_1^{m_1+1} \rho_2^{m_2+1} (\rho_1-\rho_2)^{m_0} -$$

$$\begin{aligned}
 & - \rho_2 (1-\rho_2) (\rho_1-\rho_2) \rho_2^{m_2+m_0} (1-\rho_1^{m_1+1}) - \\
 & - \rho_2 (1-\rho_2) (1-\rho_1) \rho_2^{m_2} \rho_1^{m_1+1} (\rho_1^{m_0}-\rho_2^{m_0}) \\
 \rho_2 (N_2 - T_2) = & (1-\rho_1^{m_1+m_0+1}) (\rho_1 \rho_2 - \rho_1 \rho_2^{m_2+2} - \rho_2^2 + \rho_2^{m_0+3}) + \\
 & + \rho_1 \rho_2^{m_2+2} - \rho_1 \rho_2^{m_2+m_0+2} - \rho_2^{m_2+3} + \rho_2^{m_2+m_0+3} - \\
 & - \rho_1^{m_1+m_0+1} \rho_2^{m_2+2} + \rho_1^{m_1+1} \rho_2^{m_2+m_0+1} + \rho_1^{m_1+m_0+1} \rho_2^{m_2+3} - \\
 & - \rho_1^{m_1+1} \rho_2^{m_2+m_0+3} - (1-\rho_1^{m_1+1}) \rho_1 \rho_2^{m_2+m_0+1} + (1-\rho_1^{m_1+1}) \rho_2^{m_2+m_0+2} \\
 & + (1-\rho_1^{m_1+1}) \rho_1 \rho_2^{m_2+m_0+2} - (1-\rho_1^{m_2+1}) \rho_2^{m_2+m_0+3} - \\
 & - (1-\rho_1) \rho_1^{m_1+m_0+1} \rho_2^{m_2+1} + (1-\rho_1) \rho_1^{m_1+1} \rho_2^{m_2+m_0+1} + \\
 & + (1-\rho_1) \rho_1^{m_1+m_0+1} \rho_2^{m_2+2} - (1-\rho_1) \rho_1^{m_1+1} \rho_2^{m_2+m_0+2}
 \end{aligned}
 \tag{B.32}$$

$$\begin{aligned}
 N_2 = & (1-\rho_1^{m_1+m_0+1}) (\rho_1 - \rho_2 - \rho_1 \rho_2^{m_2+1} + \rho_2^{m_2+2}) + \\
 & + \rho_1 \rho_2^{m_2+1} - \rho_1 \rho_2^{m_2+m_0+1} - \rho_2^{m_2+2} + \rho_2^{m_2+m_0+2} - \\
 & - \rho_1^{m_1+m_0+1} \rho_2^{m_2+1} + \rho_1^{m_1+1} \rho_2^{m_2+m_0+1} + \rho_1^{m_1+m_0+1} \rho_2^{m_2+2} - \\
 & - \rho_1^{m_1+1} \rho_2^{m_2+m_0+2}
 \end{aligned}
 \tag{B.33}$$

The utilization of transmission link 2 when $\rho_2 \rightarrow \infty$ equals

$$\lim_{\rho_2 \rightarrow \infty} \rho_2 \left(\frac{N_2 - T_2}{N_2} \right) \quad (\text{B.34})$$

Let us start with $m_0 > 0$.

In both the numerator and the denominator the highest power of ρ_2 is $m_2 + m_0 + 2$ as the sum of the coefficients for $\rho_2^{m_2 + m_3 + 3}$ in the denominator equals zero.

$$1 - \rho_1^{m_1 + 1} - (1 - \rho_1^{m_1 + 1}) = 0 \quad (\text{B.35})$$

The coefficient for $\rho_2^{m_2 + m_3 + 2}$ in the denominator is

$$- \rho_1 + \rho_1^{m_1 + 1} + (1 - \rho_1^{m_1 + 1}) + \rho_1 (1 - \rho_1^{m_1 + 1}) - (1 - \rho_1) \rho_1^{m_1 + 1} \quad (\text{B.36})$$

which is equal to

$$1 - \rho_1^{m_1 + 1} \quad (\text{B.37})$$

The coefficient for $\rho_2^{m_2 + m_3 + 2}$ in the numerator is

$$1 - \rho_1^{m_1 + 1} \quad (\text{B.38})$$

Thus

$$\lim_{\rho_2 \rightarrow \infty} \rho_2 (1 - B_2) = 1 \quad (\text{B.39})$$

Now let $m_0 = 0$

In both the numerator and the denominator the highest power of ρ_2 are m_2+2 as the sum of the coefficients for $\rho_2^{m_2+3}$ in the denominator equals zero.

The coefficients for $\rho_2^{m_2+3}$ in the denominator are

$$(1-\rho_1^{m_1+1}) - 1 + 1 + \rho_1^{m_1+1} - \rho_1^{m_1+1} - (1-\rho_1^{m_1+1}) \quad (\text{B.40})$$

The coefficients for $\rho_2^{m_2+2}$ in the denominator are

$$\begin{aligned} & -(1-\rho_1^{m_1+1})\rho_1 + \rho_1 - \rho_1 - \rho_1^{m_1+1} + \rho_1^{m_1+1} + \rho_1^{m_1+1} + \\ & + (1-\rho_1^{m_1+1}) + \rho_1(1-\rho_1^{m_1+1}) + (1-\rho_1)\rho_1^{m_1+1} - (1-\rho_1)\rho_1^{m_1+1} \end{aligned} \quad (\text{B.41})$$

which is equal to

$$+(1-\rho_1^{m_1+1}) \quad (\text{B.42})$$

and those for $\rho_2^{m_2+2}$ in the numerator are

$$(1-\rho_1^{m_1+1}) - 1 + 1 + \rho_1^{m_1+1} - \rho_1^{m_1+1} = (1-\rho_1^{m_1+1}) \quad (\text{B.43})$$

which gives the same limit as in the case when $m_0 > 0$.

Thus we have found the asymptotic values for the utilization under the unbalanced load situation to be

a) CS

utilization of transmission link 1 as $\rho_2 \rightarrow \infty$

$$\lim_{\rho_2 \rightarrow \infty} \rho_1 (1-B) = 0$$

and that of transmission link 2

$$\lim_{\rho_2 \rightarrow \infty} \rho_2 (1-B) = 1$$

b) SMA

utilization of transmission link 1 as ρ_2

$$\lim_{\rho_2 \rightarrow \infty} \rho_1 (1-B_1) = \frac{\rho_1 (1-\rho_1^{m_1})}{(1-\rho_1^{m_1+1})}$$

and that of transmission link 2

$$\lim_{\rho_2 \rightarrow \infty} \rho_2 (1-B_2) = 1$$

REFERENCES

- [1] N Abramson, "The Aloha System - Another Alternative for Computer Communications", AFIPS Conference. Proc, FJCC 1970.
- [2] A R Adam, "The BPO National Packet Switched Service", Conf on Communications Equipment and Systems, IEE, London, April 1980.
- [3] S E Andersen, "On the Fluctuations of Sums of Random Variables", Math Scand, vol 2, 1954.
- [4] P Baran, "On Distributed Communications", series of eleven reports, RAND Corporation, Aug 1964.
- [5] F Baskett, K M Chandy, R R Muntz and J Palacios, "Open, Closed and Mixed Networks with Different Classes of Customers", Journal of the ACM, vol 22, no 2, 1975.
- [6] V E Benes, General Stochastic Processes in the Theory of Queues, Addison-Wesley, 1963.
- [7] D R Boggs and R M Metcalfe, "Ethernet: Distributed Packet Switching for Local Computer Networks", Communications of the Association for Computing Machinery, vol 19, no 7, 1976.
- [8] F Borgonovo and L Fratta, "A Model for Finite Storage Message Switching Networks", Istituto di Elettronica and Centro Telecomunicazioni Spaziati - CNR Politecnico di Milano, Italy.

- [9] P J Burke, "The Output of a Queueing System", Operations Research, vol 4, 1966.
- [10] M Buttò, G Colombo, G Taggiasco and A Tonietti, "Models for the Performance Evaluation of a Packet Switching Network with Retransmission Time-out", CSELT Rapporti tecnici, vol VI, no 1, March 1980.
- [11] J P Buzen, "Computational Algorithms for Closed Queueing Networks with Exponential Servers", Communications of the ACM, vol 16, no 9, 1973.
- [12] J P Buzen, "Queueing Network Models of Multi-programming", PhD Thesis, Div Eng and Applied Physics, Haward Univ Cambridge, Mass, May 1971.
- [13] J P Buzen, "Cost Effective Analytic Tools for Computer Performance Evaluation, Proc IEEE COMPCON 1975, IEEE, New York.
- [14] V G Cerf and R E Kahn, "A Protocol for Packet Network Intercommunication", IEEE Trans on Communications, vol 22, no 5, 1974.
- [15] V G Cerf and P T Kirstein, "Issues in Packet Network Interconnection, Proc of the IEEE, vol 66, no 11, 1978.
- [16] K M Chandy and C H Sauer, "Approximate Methods for Analyzing Queueing Network Models of Computing Systems", Computing Surveys, vol 10, no 3, 1978.

- [17] W W Chu, "A Study of Asynchronous Time Division Multiplexing for Time-Sharing Computer Systems", AFIPS Conference Proc, FJCC, Las Vegas, Nevada, 1969.
- [18] W W Clipsham, F E Glave and M L Narraway, "Datapac Network Overview", Proc of ICCC, 1976.
- [19] J W Cohen, The Single Server Queue, North Holland, 1969.
- [20] D Cohen, "On Packet Speech Communication", proc of the ICCC, Atlanta, 1980.
- [21] D R Cox, Renewal Theory, Chapman and Hall Science Paperbacks, London 1970 (first published 1962).
- [22] D R Cox and W L Smith, Queues, Methuen, London, 1958.
- [23] D R Cox and H D Miller, The Theory of Stochastic Processes, Chapman and Hall, Science Paperbacks, London, 1977 (first published in 1965).
- [24] A Danet, R Depres, A Lerest, G Pichon and S Ritzentaler, "The French Public Packet Switching Service: The TRANSPAC network", Proc of ICCC, Toronto, Aug 1976.
- [25] D W Davies and D L A Barter, Communication Networks for Computers, Wiley, London, 1973.
- [26] G J Foschini, B Gopinath and J F Hayes, "Optimum Allocation of Servers to Two Types of Competing Customers", IEEE Trans on Communications, vol 29, no 7, 1981.

- [27] P D Finch, "The Output Process of the Queueing System M/G/1", Journal of Royal Statistical Society, Ser B, vol 21, no 2, 1959.
- [28] H Frank and W Chou, "Topology Optimization of Computer Networks", Proc of the IEEE, vol 60, no 11, 1972.
- [29] G L Fultz, "Adaptive Routing Techniques for Message Switching Computer Communications Networks", School of Engineering and Applied Science, University of California, Los Angeles, UCLA-ENG-7252, July 1972.
- [30] M Gerla, "The Design of Store-and-Forward (S/F) Networks for Computer Communications", School of Engineering and Applied Science, University of California, Los Angeles, UCLA-ENG-7319, Jan 1973.
- [31] M Gerla and L Kleinrock, "Flow Control: A Comparative Survey", IEEE Trans on Communications, vol 29, no 4, 1980.
- [32] T Giammo, "Validation of a Computer Performance Model of the Exponential Queueing Network Family", Acta Inf, vol 7, 1976.
- [33] M Gien and H Zimmermann, "Design Principles for Network Interconnection", Sixth Data Communications Symposium, Pacific Grove, California, Nov 1979.
- [34] W J Gordon and G F Newell, "Closed Queueing Systems with Exponential Servers", Operations Research, vol 15, no 2, 1967.

- [35] J L Grange, "Operating the CIGALE Packet Switching Network: Concepts, Techniques and Results", EUROCON'77, Venice, May 1977.
- [36] J L Grangé and M Irland, "Proposal for Simulation of Cigale 1976: Congestion Control", University of Waterloo, Computer Communications Networks Group, Report I-22, Oct 1975.
- [37] D E Hadley and B R Sexton, "The British Post Office Experimental Packet Switched Service (EPSS) - A Review of Development and Operational Experience", Proc of ICC, Japan, 1978.
- [38] F E Heart, R E Kahn, S M Ornstein, W R Crowther and D C Walden, "The Interface Message Processor for the ARPA Computer Network", AFIPS Conference Proc SJCC, 1970.
- [39] P H Hughes and G Moe, "A Structural Approach to Computer Performance Analysis", Proc 1973 AFIPS National Computer Conference, Vol 42, AFIPS Press, Montvale, N J.
- [40] M Irland, "Analysis and Simulation of Congestion in Packet Switched Networks", PhD dissertation, University of Waterloo, Waterloo, Canada, April 1977.
- [41] M Irland, "Buffer Management in a Packet Switch", IEEE Trans on Communications, vol 26, no 3, 1978.
- [42] M Irland, "Queueing Analysis of a Buffer Allocation Scheme for a Packet Switch", Proc IEEE National Telecommunications Conference, New Orleans, Dec 1975.

- [43] J R Jackson, "Networks of Waiting Lines",
Operations Research, vol 5, 1957.
- [44] R E Kahn, The Introduction of Packet Satellite
Communications, NTC Conf Rec, National Telecom-
munications Conference, Washington D C, Nov 1979.
- [45] F Kamoun, "Design Considerations for Large Com-
puter Communications Networks", PhD dissertation,
University of California, Los Angeles, Eng Rep
7642, April 1976.
- [46] F Kamoun, "A Drop and Throttle Flow Control
Policy for Computer Networks", IEEE Trans on
Communications, vol 29, no 4, 1981.
- [47] F Kamoun and L Kleinrock, "Analysis of Shared
Storage in a Computer Network Environment",
Proc 9th Hawaii International Conference on
System Science, Hawaii, Jan 1976.
- [48] F Kamoun and L Kleinrock, "Analysis of Shared
Finite Storage in a Computer Network Node En-
vironment under General Traffic Conditions",
IEEE Trans on Communications, vol 28, no 7,
1980.
- [49] A Kaul, "A Model for a Packet Switching Node
Including Packet Retransmission Effects",
Comsat Technical Review, vol 9, no 2B, 1979.
- [50] J F C Kingman, "On the Algebra of Queues",
Journal of Applied Probability, vol 3, 1966.
- [51] L Kleinrock, Queueing Systems, Volume I: Theory,
Wiley Interscience, New York, 1975.

- [52] L Kleinrock, Queueing Systems, Volume II: Computer Applications, Wiley, New York, 1976.
- [53] L Kleinrock, Communication Nets, Stochastic Message Flow and Delay, McGraw-Hill Inc, N Y, 1964.
- [54] L Kleinrock, "Principles and Lessons in Packet Communications", Proc of the IEEE, vol 66, no 11, 1978.
- [55] L Kleinrock, W E Naylor and M Opderbeck, "A study of Line Overhead in the Arpanet", Communications of the ACM, vol 19, no 1, 1976.
- [56] H Kobayashi, Modelling and Analysis, an Introduction to System Performance Evaluation Methodology, Addison-Wesley Publishing Company, 1978.
- [57] U Körner, "Analysis of Shared Storage in a Computer Communication Network, Ninth International Teletraffic Congress, Spain, 1979.
- [58] J Labetoulle and G Pujolle, Modelling of Packet Switching Communication Networks with Finite Buffer Size at Each Node, Computer Performance, K M Chandy and M Reiser (Eds), North Holland Publishing Company, 1977.
- [59] S S Lam, "Store and Forward Buffer Requirements in a Packet Switching Network", IEEE Trans on Communications, vol 24, no 4, 1976.
- [60] S S Lam, "Queueing Networks with Population Size Constraints". IBM Journal of Research and Development, vol 21, no 4, 1977.

- [61] G Latouche, "Exponential Servers Sharing a Finite Storage: Comparison of Space Allocation Policies", IEEE Trans on Communications, vol 28, no 6, 1980.
- [62] S S Lavenberg, "The Steady State Queueing Time Distribution for the M/G/1 Finite Capacity Queue", Management Sci, vol 21, 1975.
- [63] L Lipsky and J D Church, "Applications of a Queueing Network Model for a Computer System", Computing Surveys, vol 9, no 3, 1977.
- [64] J D C Little, "A Proof of the Queueing Formula $L = \lambda W$ ", Operations Research, vol 9, 1961.
- [65] Logica Ltd, "Eurodata '79 - a Study of Data Communications in Western Europe 1979-1987", Logica Ltd, 1980.
- [66] P H Maier, "Analysis of Packet Switching Communication Networks with Finite Buffers", FREQUENZ 33, 7/8, 1979.
- [67] P H Maier, "Small Queueing Networks with Finite Storage Capacities", Proc of the Eleventh Annual Southeastern Symposium on System Theory, Clemson, S C, March 1979.
- [68] R R Muntz, "Queueing Networks: A Critique of the State of the Art and Directions for the Future", Computing Surveys, vol 10, no 3, 1978.
- [69] M C Pennotti and M Schwartz, "Congestion Control in Store-and-Forward Tandem Links", IEEE Trans on Communications, vol 23, no 12, 1975.

- [70] L Pouzin, "Methods, Tools and Observations on Flow Control in Packet Switched Data Networks", IEEE Trans on Communications, vol 29, no 4, 1981.
- [71] L Pouzin, "Flow Control in Data Networks - Methods and Tools", Proc of ICCC, Toronto, 1976.
- [72] L Pouzin, "Presentation and Major Design Aspects of the CYCLADES Computer Network", 3rd Data Com Symp, Tampa, Florida, Nov 1973.
- [73] L Pouzin, "CIGALE, the Packet Switching machine of the CYCLADES Computer Network", IFIP Congress, Stockholm, Aug 1974.
- [74] L Pouzin and H Zimmermann, "A Tutorial on Protocols", Proc of the IEEE, vol 66, no 11, 1978.
- [75] W L Price, "Simulation of a Packet Switched Data Network Operating under Isarithmic Control with a Revised Link and Node Protocol", NPL Report COM 71.
- [76] M Reiser and H Kobayashi, "The Effects of Service Time Distributions on System Performance", Proc of the IFIP Congress, Stockholm, 1974.
- [77] M Rich and M Schwartz, "Buffer Sharing in Computer Communication Network Nodes", IEEE Trans on Communications, vol 25, no 9, 1977.
- [78] L G Roberts, "The Telenet Network: the Benefits of Public Packet Service", Proc of ICCC, Toronto, 1976.

- [79] L G Roberts, "The Evolution of Packet Switching",
Proc of the IEEE, vol 66, no 11, 1978.
- [80] C A Rose, "Measurement Procedure for Queueing
Network Models of Computer Systems", Computing
Surveys, vol 10, no 3, 1978.
- [81] H Rudin, " An Introduction to Flow Control",
Proc of the ICC, Toronto, 1976.
- [82] H Rudin, "Congestion Control: Preview and some
Comments", IEEE Trans on Communications, vol 29,
no 4, 1981.
- [83] T L Saaty, Elements of Queueing Theory with App-
lications, McGraw-Hall, New York, 1961.
- [84] R A Scantlebury and P T Wilkinson, "The National
Physical Laboratory Data Communication Network",
Proc of ICC, Stockholm, 1974.
- [85] P J Schweitzer and S S Lam, "Buffer Overflow in
a Store-and-Forward Network Node", IBM Journal
of Research and Development, vol 20, no 6, 1976.
- [86] P J Schweitzer and A Konheim, "Buffer Overflow
Calculations Using an Infinite Capacity Model",
Stochastic Processes and their Applications,
vol 6, no 3, 1978.
- [87] D E Sproule and F Mellor, "Routing, Flow and
Congestion Control in the Datapac Network",
IEEE Trans on Communications, vol 29, no 4,
1981.

- [88] C A Sunshine, "Interconnection of Computer Networks", Computer Networks, vol 1, no 3, 1977.
- [89] L R W Tymes, "Routing and Flow Control in Tymnet", IEEE Trans on Communications, vol 29, no 4, 1981.
- [90] H Zimmermann, "The Cyclades Experience. Results and Impacts", IFIP Congress, Toronto, Aug 1977.

